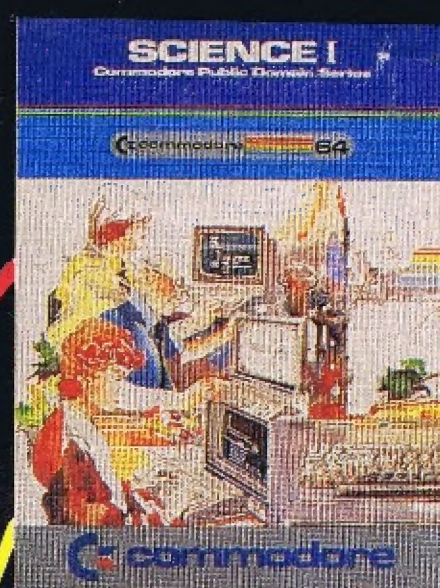
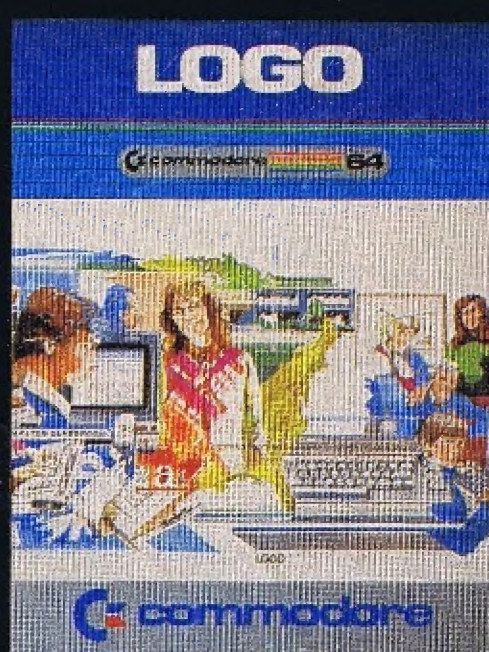
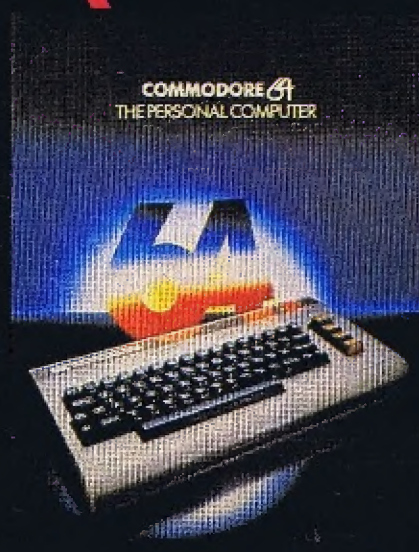
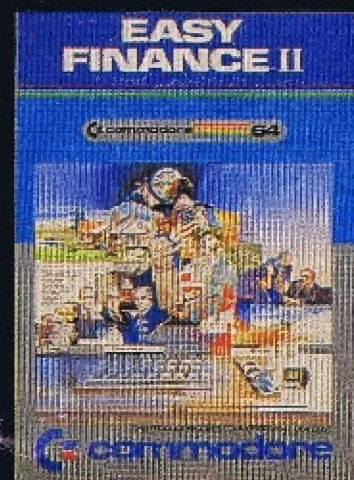
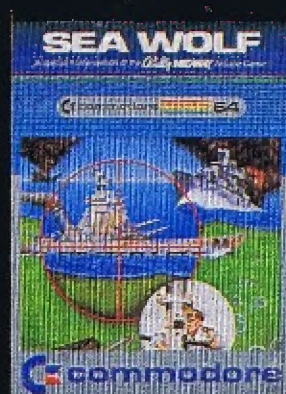
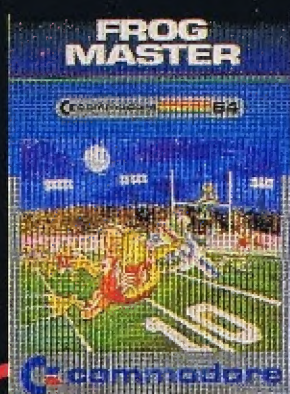
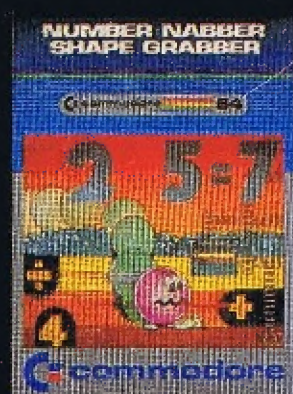
VIC! THE MAGAZINE FOR COMPUTER AGE

with Commodore 64

特集：ひとめでわかるコモドール64の全貌

Vol. 11

500YEN

￥34,800

遊べば遊ぶほど、親しくなる。知れば知るほど、知りたくなる。いつも一緒にいたくなる。もう片時も離れたくない。コンパクトながら中身は大型、本格派。あのコモドールのマックスマシーン。少年よ、遊びの天才たれ！

キャンペーンセール実施中

★新ゲーム登場!! ●ウィザードオブウォー ●キックマン 各￥2,800 続々ゲームシリーズ登場

●お求めは全国百貨店・スーパーの玩具売場及び玩具専門店で。

■ゲームカートリッジ各￥2,800…●アヴェンジャー●ジュピターランダー●レーダーラットレース●ロードレース●モールアタック●スーパーエイリアン●クラウンズ●マネーウォーズ●オメガレース ■ミニベーシック ￥3,800 ■ジョイスティック ￥3,800 ■パドル ￥4,800

製造元 コモドール・ジャパン株式会社 ●東京都港区東新橋2-12-7住友東新橋ビル2号館 東京本社〒105☎03(433)6111(大代表)

総発売元 株式会社ムーミン ●東京都港区芝4-7-7 田町アネックス 東京本社〒108☎03(455)6161(代表)●大阪市南区長堀橋筋1-45-1日生長堀橋ビル大阪支店〒542☎06(251)4001(代表)

C O N T E N T S

VIC!

THE MAGAZINE FOR COMPUTER AGE

発行／コモドール・ジャパン株式会社 広報室 発行人兼編集人／牛田財司

VIC!SPECIAL

特集

新・解体新書

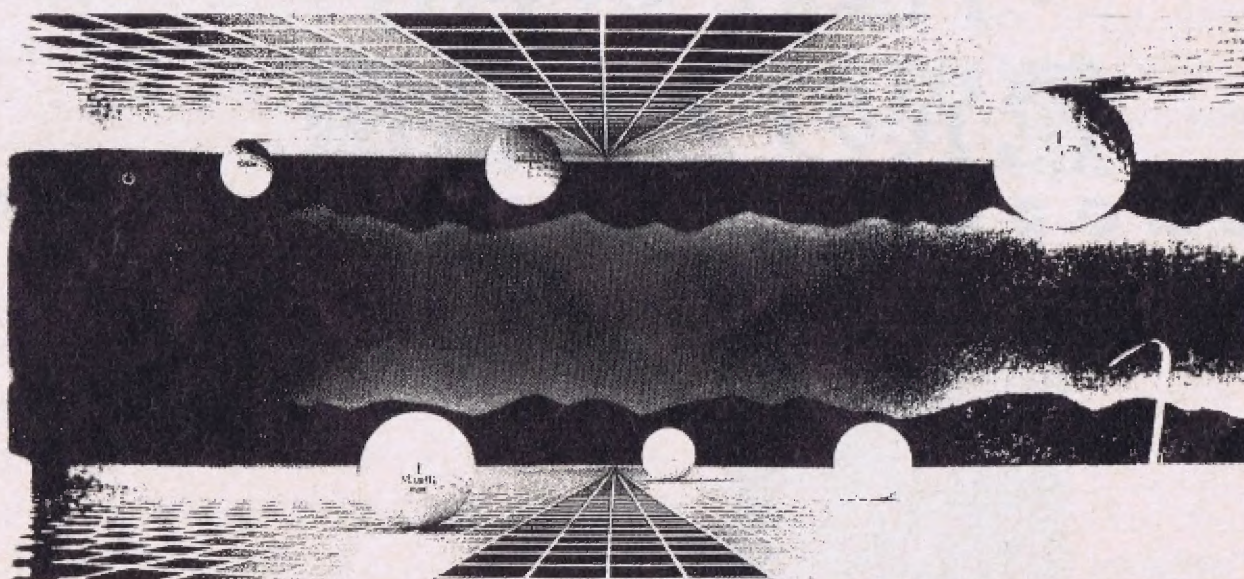
ひと目でわかる、コモドール64の全貌。

男の道具は、ひたすら道具に徹しなさい。
「誰にも触れさせたくない。コンピュータに縛られたくない。」そんなコンピュータ感覚が、ニーズとして浸透してきました。○○派 ○○族と言ったジャンルを超えた大人の感性、そんな「自分らしさ」を楽しむ人たちに届けます。
COMMODORE 64!!
自分を変身させるコンピュータ!!

First Step by Commodore 64

フロンティア・スピリッツを携え、常にワールドワイドな躍進をつづけるコモドール。その豊かな足跡の証しとして、'82年にはパソコンの最高名誉賞"Computer of the Year"を受賞。これにより、アメリカ3大パーソナルコンピューターメーカーの名声は揺るぎないものとなりました。そのユニークかつ広角的コンセプトをもつコモドールからユーザーへのメッセージ、それは"人とコンピュータとの対話(コミュニケーション)"。メッセージを機械(メカ)に託し、コモドール64は誕生しました。コンピュータと本気で取組みたい人、しかし不安…という人にこそ、コモドール64は格好のツール。BASICを鮮明に理解できるマニュアル付き。しかし、コミュニケーションはもっとハイレベルを求めます。たとえば、ゲームを自分でプログラミングしたい時、BASICでは不自由です。コンピュータの内部(インナー)を知ってこそ素晴らしいゲームも創造できるのです。

Basic is best communication

だからコンピュータを始める最短手段として、そして次のステップへ向かうコンピュータの中身を知りつくす手段として、コモドール64。本当にコンピュータを知りたい人のために送ります。先進技術を凝縮したコモドールから、自信をもって、いま——。

VIC!SPECIAL

コモドール64は、BASICを自由に使いこなせたら自然に内部も解ってくる特徴をもっています。それは、次の段階に進む時、たとえばマシン語をマスターしようとする場合、非常に大きなメリットです。

VICII CHIP

GRAPHICS

映像人間・感涙。複雑なテクニックぬきのスプライト機能で、超迫力グラフィックスをプロデュース。自由自在にC・G(コンピュータグラフィックス)をフィンガーペインティングする、コモドール64。

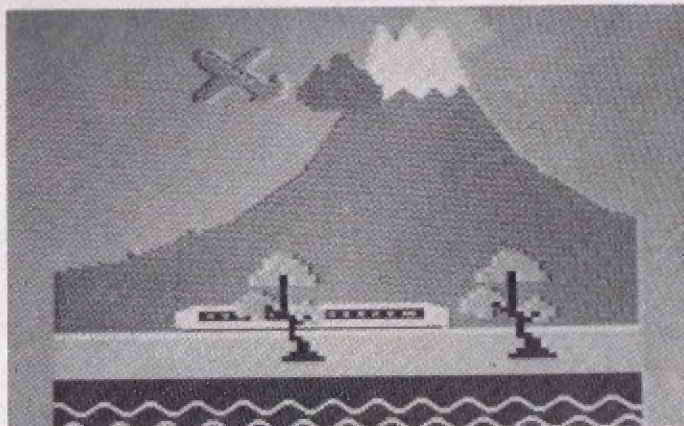
■不安定だったBASICでの図形移動が、スプライト機能でスムーズ&スピーディ。

■画面に奥行きがでるので、立体感のあるグラフィックが期待できる。

■スプライトどうしの衝突を簡単に検出でき、ゲーム作成に大きなメリット。

■左右、上下に2倍の拡大が可能。迫力満点の妙味。

■色を重視し、画面と独立したカラーの指定が可能。

スプライト機能

キャラクターデータを設定しておくだけで、いつでも画面のどこにでもピクチャーを出すことができる。スプライトvsスプライト、スプライトvsバックグランドの衝突も簡単に検出。64種、最大128種のキャラクター設定可能。

SID CHIP

MUSIC

音楽人間、待望。オリジナル・サウンドをらくらくメーキング。最少のプログラムで無限大のアビリティを創造する、気軽なミュージック・シンセサイザー、コモドール64。

■シンセサイザーを内包してしまった、驚異のサウンド/ミュージック機能。

■SID(サウンド・インターフェイス・ディバイス)採用により、3声・9オクターブのワイドな音域演奏。

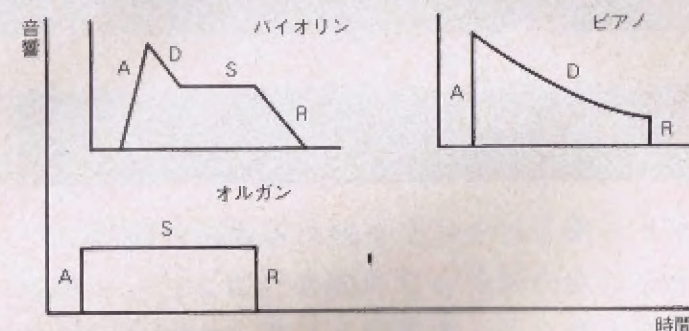
■波形①、エンヴェロープ②、パルス幅、周波数を自在に駆使して、オリジナルトーンが創れます。

■ゲームの効果音づくりは、まさにプロ級。

※① 64のSIDの3声は波形を制御するコントロール・レジスタを持ち、三角波・のこぎり波・可変パルス波・ノイズの4波形を選択できます。

波形	略図	印象	楽器の例
三角波		柔かくて丸い音	フルート
のこぎり波		明るい音	バイオリン・チェロなどの弦楽器
可変幅パルス波	短形波(b/a-1/2)	明るくうつろな音	クラリネット
	非対称パルス波(b/a-3/4)	鼻声がかった音	オーボエなどのリード楽器
ノイズ		雑音	小太鼓・シンバル・風の音

※② 音量の時間的変化を示すもの。その変化は、アタック・ディケイ・サスティン・リリースに分けられる。このエンヴェロープをプログラミングできるので、楽器のシミュレートがもちろんOKです。

●エンヴェロープ例(バイオリン、ピアノ、オルガン)

コモドール64がビッグメカと呼ばれる理由 驚異の2大特徴。

COMMODORE 64 CHARACTER

Basic is best communication

MANUAL

総380ページ。BASIC言語の解説のみにとどまらず、プログラムの組立て、テクニックなどビギナーがつまずき易い箇所にポイントをおいた手引書。

INTRODUCTION TO BASIC

BASICのマスターは対話式でクイック＆スムーズ、そして完璧。

テキスト：パソコンのA to Zとプログラム作成が簡明に解かれています。
ソフトカセット：テキストに即したデモソフト(サウンドカラー・グラフィックス・スプライト機能を含む)や、理解しやすい対話式クイズソフト、ドリルソフトなど全17のプログラムを収めたカセット2本。
テンプレート：フローチャート作成用。

セット価格 ¥9,800

COMPONENT SYSTEM

コモドール64を自在に駆使させるワイドな周辺機器ラインアップ

1011A
RS232Cアダプター・カートリッジ(ターミナル・タイプ)

1011B
RS232Cアダプター・カートリッジ(カレント・ループ・タイプ)

ユーザー・ポートに接続することにより、RS232Cターミナル・タイプ・インターフェイス(1011A)、RS232C カレント・ループ・インターフェイス(1011B)をもった各種機器が接続可能になります。また、音響カプラの接続で、電話線を使って遠く離れたコンピュータどうしのコミュニケーションも可能。

1701
高解像度カラー モニター(モニターケーブル付き・2ピン)

●コンポジット入力(通常のビデオ入力が可能)●音声回路および14インチスピーカー内蔵●コンポジット出力を同期信号とカラー信号とにセパレート入力できるため、より鮮明な画像が可能(専用の3ピンモニターケーブルを使用した場合)●通常の家庭用モニターケーブルとして使用可能●付属のモニターケーブルでVIC-1001にも接続可能。

1525J
グラフィック・プリンタ

シリアル・ポートよりダイレクトに接続。ドット・マトリックス・インパクト・プリンタ。1文字8×8ドット構成により表示。C-64のもつすべての文字(英数字、グラフィック文字、カタカナ)を印字可能。プログラムによるキャラクターの設定可能。

C2N
カセット・ドライブ

記憶容量、約160Kバイト(C-30使用時)。記憶方式はコモドール方式(500ボー)。C-64のカセット・インターフェイスに接続。SAVE中のランプ装備付。

1541
シングル・フロッピー・ディスク・ドライブ

シリアル・バスに付属ケーブルでダイレクトに接続(接続ケーブル付)。インテリジェント・フロッピー・ディスクでC-64のメモリーを使いません。小型ながら170Kバイトものデータを高速処理。大容量ファイルとして使用でき、カセットベースに比べて飛躍的なデータ処理が可能。使用ディスケット:5.25インチ標準ミニ・フロッピー(ソフト・セクター)記憶容量175Kバイト/ディスケット ディレクトリー・エントリー:144/ディスケット セクター/トラック:17〜21バイト/セクター:256 トラック:35 ブロック:683。

1311/1312
ジョイスティック/パドル

ゲームをいっそうゲームらしくできるツール。

プログラマーズ・エイド
☆近日発売

BASICのプログラミング、プログラムのデバックを大幅に効率化する、他に類をみない“TOOLKIT”。プログラミングおよびデバック(エディト)用の各種BASICコマンドを追加。RENUMBER. MERGE, FIND, CHANGE, DELETE, AUTO, TRACE, STEP, DUMP, HELPなど。ファンクションキーに2モード計24種のコマンドをアサインしています(ユーザーが変更することも可能)。コントロール・ファンクションを追加。カートリッジスロットに接続。

マシン・ランゲージ・モニター
☆近日発売

C-64でのマシン語プログラムの作成、デバックを大幅に効率化するツールキットです。①キーボードからマシン・ランゲージ・コマンドを入力することによりCPUの動きをコントロールでき、システムをファイルできます。②CPUのステップ動作が可能③レジスタ、プログラムステータス、スタックポインタおよびメモリの内容を自由に変更できます。④6500系のミニアセンブラーが使えます。

スーパー・エクスパンダー(V・S・P)

V・S・Pはハイレゾリューション・グラフィック・コマンド、スプライト・コマンド、ミュージックパッケージ、その他からなる。C-64の機能をフルにいかしたBASIC拡張パック。カートリッジスロットに接続。追加されるBASICコマンド：COLOR, GRAPHIC, LOCATE, RBUMP, RCLR, RDOT, RGR, RJOY, RPEN, RPOT, SCALE, BOX, CIRCLE, CHAR, DRAW, GSHAPE, TAINT, SSHAPE, COLINT, DEFSPR, MOVSPR, RSPCOL, RSPPOS, RSPR, SPRCOL, SPRITE, SPRSAV, その他ミュージック関係、ファンクションキー関係など。

☆近日発売

●COMMODORE 64のスペック

項目		仕様
CPU		標　　準　6510（6502ファミリー） オプション　Z80（カードにより供給）
メモリ	ROM	20Kバイト（BASIC・OS16K、キャラクタージェネレータ4K）
	RAM	64Kバイト
表示	テキスト表示	40桁×25行
	グラフィック表示	320×200ドット（ノーマル・モード） 160×200ドット（マルチカラー・モード）
	カラー表示	ボーダー／スクリーン／キャラクターを各16色選択可
	スプライト表示	画面とは独立に移動できる物体を8個同時に表示可能
キーボード		フルサイズ・タイプライター・キーボード（JIS配列準拠） プログラマブル・ファンクションキー カーソル・コントロールキー
ディスプレイ		家庭用TVまたは、専用モニター
シリアルインターフェイス		実装（ディスク・ドライブ、プリンタを接続）
カセットインターフェイス		実装（専用カセット・ドライブを接続）
ユーザーポート		実装（8ビット入出力兼用）
コントロールポート		2個実装（ジョイスティク、パドルを接続）
RS-232Cインターフェイス・カートリッジ		ユーザーポートに接続可
IEEEインターフェイス・カートリッジ		予定
サウンド機能		6581（SID＝サウンド・インターフェイス・ディバイス） 3重和音＋1外部音声入力 音域9オクターブ（0.061Hz〜3.9kHz。0.061Hzきざみ） 4波形（三角波、のこぎり波、可変幅パルス波、ノイズ） プログラマブル・ADSR（アタック、ディケイ、サスティン、リリース） プログラマブル・フィルター（ハイパス、ローパス、バンドパス、ノッチパス） オーディオ出力：DIN8ピンコネクタで接続可
CRTコントローラー		6567（VIC＝ビデオ・インターフェイス・チップ）
言語		BASIC（標準実装） LOGO（予定）・UCSD PASCAL（予定）・PILOT（予定） 6510　アセンブリ・ランゲージ
電源		AC 100V、50／60Hz
外形寸法		404(W)×204(D)×74(H)mm　重量 1.8kg

※仕様は予告なく変更する場合があります。

MGM/UA映画＝CIC配給　東宝系公開

ビ〜ンカ〜ン
コンピュータ映画「WAR GAMES」

パソコン少年の無邪気さがひきおこす第3次世界大戦!?ひとごとじゃなくゾクッ、とする「WAR GAMES(ウォー ゲーム)」。'84年正月に日本上陸。

突如、大スクリーンにうつし出された第3次世界大戦迎撃プログラム！第一撃（先制）はソ連。目標は米ラスベガスとシアトル。続いてアメリカによる第二撃（報復）。まるで、夜空の花火のように点滅する核攻撃の表示。それは、間違いなく世界の破滅を意味していた！——。

思わぬ有事の突発にあわてふためく、ここコロラド山中NORAD(ノーラッド)（北米大陸防空軍）司令部。システムの故障か？ソ連の謀略か？否、それがパソコン狂いの高校生による単なる遊び心から発したものだとは、その時、誰も知るよしはなかった——！

本年度カンヌ映画祭クロージング・ナイトに特別招待作品として上映され、あまりのリアリティ、恐怖感に轟々たる反響を呼んだ凄いエキサイティング・ストーリーがやってきた。その名は「ウォー・ゲーム」。

主人公デビッドはコンピュータにやたら強いハイティーン。ある日、新種のビデオ・ゲームの内容を探ろうと、自分のパソコンを操作するうち、正体不明のプログラムと接触した。〃世界全面核戦争〃というそのゲームの面白いこと。同じ頃核戦争ぼっ発の表示に大騒ぎのNORAD司令部。なんと少年のパソコンは、国防省の核戦略プログラムと互いにインプットし合っていたのだ！

無邪気な少年のファンタジー・ワールドが、恐怖の現実へ暗転し、歯止めのきかない世界全面核戦争へ突入する—。溢れる臨場感に試写を観た専門家は口を揃えて〃限りない可能性〃を指摘。コンピューター万能時代に警鐘を打ち鳴らす傑作サスペンスだ。

監督…あのジョン・トラボルタの「サタデー・ナイト・フィーバー」や話題作「ブルー・サンダー」などの新鋭、ジョン・バダム。
撮影…「天国から来たチャンピオン」「シャーキーズ・マシーン」のウィリアム・A・フレイカー。
音楽…「この生命誰のもの」のアーサー・B・ルビンシュタイン。
製作総指揮…人気TV番組「チャーリーズ・エンジェル」「刑事スタスキー＆ハッチ」などを生みだした才人、レナード・ゴールドバーグ。
'83年夏、「スターウォーズ・ジェダイの復讐」「フラッシュダンス」と並び全米を制したヒット作。

テープファイルの記録方式。

カセット・テープの記録方式には一般に、カンサスシティ・スタンダード方式、FSK方式などが使われますが、コモドールでは独自の記録方式を使用しています。以下はこのコモドール方式について、最新版の『1530 DATASETTE UNIT USER'S MANUAL』から抜粋したもの。しっかり読んでマスターしよう！

コモドール・コンピュータでは、付加されるカセット・ユニットのモータの動きをコンピュータが制御できるように設計されています。また、PLAY(再生)、REWIND(巻戻し)、F.FWD(早送り)のボタンが押されたことをコンピュータがセンスできるようになっています。これはカセット・ユニットに取りつけられたスイッチ1つでおこなわれています。ただし、1つのスイッチで3つのボタンをまとめてセンスしているために、3つのボタンのいずれの1つを押しても、コンピュータはボタンが押されたものと解釈して、それに従って反応します。したがって、ボタン操作には注意しなくてはなりません。

また、カセット・ユニットに使用されているメカニズムの形式上、録音、再生、巻戻し、早送り、停止、テープの出し入れなどのボタン操作は、ユーザーが手動でおこなわなければなりません。一度、適当なボタン操作によってモータを正しくスタートさせれば、その後のテープ走行はコンピュータによってすべて制御されます。

ユーザーは、コンピュータによって表示されるメッセージによって、いつPLAYあるいはRECORDを開始しなければならないのか知ることができるようになっています。しかし、巻戻し、早巻りをおこなうべき時期については、プログラムによって、指示が表示されるようにしておかなくてなりません。

PET/CBMの2つのカセット・ユニットは独立に制御されます。種々の制御ラインによってカセット＃1からのデータ読み込み、カセット＃2からのデータ読み込み、カセッ＃1、＃2のモータ・コントロール、および共通の書き込みがおこなえます。

●記録方式

コモドール・コンピュータのテープファイルに使用されているデータ構造は、最大のメモリ利用効率と最高の信頼性を保証しています。

テープへの記録をおこなった時、記録データは3種のオーディオ周波数によって表現されるデジタル録音方式により記録されます。テープファイルは、いくつかのブロックから構成され、それぞれのブロックはすべて繰りかえして記録されます。この方法により、ほとんどのオーディオ・ドロップアウトは検出可能です。さらにセカンド・ブロック(繰りかえされる2つのブロックのうちの2番目)に記録された冗長データを使い、オペレーティング・システムによって、オーディオ・ドロップアウト・エラーのほとんどが訂正されるようになっています。

読み取り動作時にはテープスピードに対して自動的な合わせ込みがおこなわれています。これは(a)カセット・ユニットの録音速度は必ずしも一定ではない、(b)テープごとにその走行特性が異なるといった事項を補正するためにおこなわれています。

合わせ込みは、テープ上の各ブロックの間で確実に同期を取るという方法でおこなわれます。このために各ブロック間には単一音からなる同期用トーンが書き込まれます。この信号はテープの位置と速度の双方を同期するのに用いられます。また、テープファイルの開始点とブロック間では同期用トーンの長さが変えてあります。各テープファイルの始めの部分には約10秒間、同期用トーンが書き込まれ、この同期用トーンによってリーダ部(テープファイルの始まりを示す記録部分)を正しく読むための自己補正動作をおこないます。一方、それぞれのブロックのあいだは、短い同期用トーンが書き込まれます。

●ブロック

ブロックには「ヘッダ・ブロック」「プログラム・ブロック」「データ・ブロック」および「エンド・オブ・テープ・ブロック」があります。各ブロックの先頭には、ブロックのタイプを表わすマーカがついています。

0　プログラム・ブロック
1　プログラム・ファイル・ヘッダ・ブロック
2　データ・ブロック
4　データファイル・ヘッダ・ブロック
5　エンド・オブ・テープ・ブロック

テープへの記録のされ方はプログラム・ファイルとデータ・ファイルの場合では、異なります。

プログラム・ファイルの場合は、ヘッダ・ブロックにつづいて、プログラム全体を記録するプログラム・ブロックがつづきます。一方、データ・ファイルの場合は、ヘッダ・ブロックにつづいて、192文字(先頭の1文字は2)ごとのいくつかのデータブロックに分けられて記録されます。カセットテープへの書き込み時にセカンダリーアドレス2を用いた場合には、プログラム・ファイルの場合でもデータ・ファイルの場合でも、最終ブロックとしてエンド・オブ・テープ・ブロックがつづきます。このエンド・オブ・テープ・ブロックは、先頭のタイプ指示バイト(5を示す)をのぞいて、ファイル・ヘッダと同一の内容です。

「ファイル·ヘッダ·ブロック」「データ·ブロック」「エンド·オブ·テープ·ブロック」のブロック本体の大きさは192文字です。プログラム·ブロックはプログラムの大きさに従ったバイト数が記録されます。

●ファイルヘッダ

ファイルヘッダ・ブロックはファイル名によるファイル管理をおこなうためにつくられたブロックです。ファイルヘッダ・ブロックは、約10秒のリーダ部のあと、テープに記録されるフィジカル·データ（実際に記録されるハードウェア寄りの意味でのデータ）のすぐ先頭にあります。

ファイルヘッダにはファイル名が書きこまれます。プログラム・ファイル名は16文字まで書きこまれます。データ·ファイル名は128文字まで書きこまれます。OPENコマンド（またはSAVE/VERIFY/LOADコマンド）では、このファイル名がサーチされ、指定のファイル名と一致するかどうかが調べられます。

●テープ・バッファ

テープ・オペレーティング・システムを設計するに当って「ファイル名の管理」とともに問題になったもう一つの点は、コンピュータからのデータ出力の速度とカセット・テープへのデータ書き込みの速度に違いがあるということです。この問題は、おのおののカセット・ドライブに1ブロックずつのメモリーをデータバッファとして割当てることによって解決されています。192文字のバッファが、カセット用にとられています。

テープ終了端が出たことを示すためにはファイルのサーチ(検索)中にこのEOTブロックが現われると、コモドール・コンピュータは自動的にテープを停め、「FILE NOT FOUND(ファイルを発見できなかった)」と表示します。一般的なマルチプル・ファイルとしては、たとえば最初にデータファイルがあり、次にプログラム·ファイルが来て、その後に「エンド・オブ·テープ」ヘッダが続く、というような場合が考えられます。

●テープファイルの動作モード

テープファイルは、2つの動作をおこないます。

(a)コンピュータからテープに書き込む。
(b)テープからコンピュータに読み込む。

(a)コンピュータからテープに書き込む場合のOPEN

流れ図Aは、書き込みを目的としてファイルをオープンした場合の、コンピュータとユーザー間のやりとりを示しています。ファイルをオープンする場合に次の2通りの場合があります。

(1)表き込みのためのOPEN。
(2)プログラムをSAVEする。

(1)と(2)とはファイルヘッダの書き方がちがいます。

テープファイルがダイレクトにオープンされた場合一つまりキーボードから直接に一には“WRITNGファイル名”(SAVINGファイル名)の表示がされます。逆にプログラムの制御によってファイルがオープンされた場合、加えて再生(PLAY)·録音(RECORD)のボタンがすでに押してあれば、スクリーンには何のメッセージも表示されません。したがって現在表示中の画面が変わってしまうこともありません。

PRINT #, CMD, SAVEコマンドによるファイルOPENのプロセス。
OPはオペレーティングシステムのこと。

(b)テープからコンピュータへの読み込みの場合のOPEN

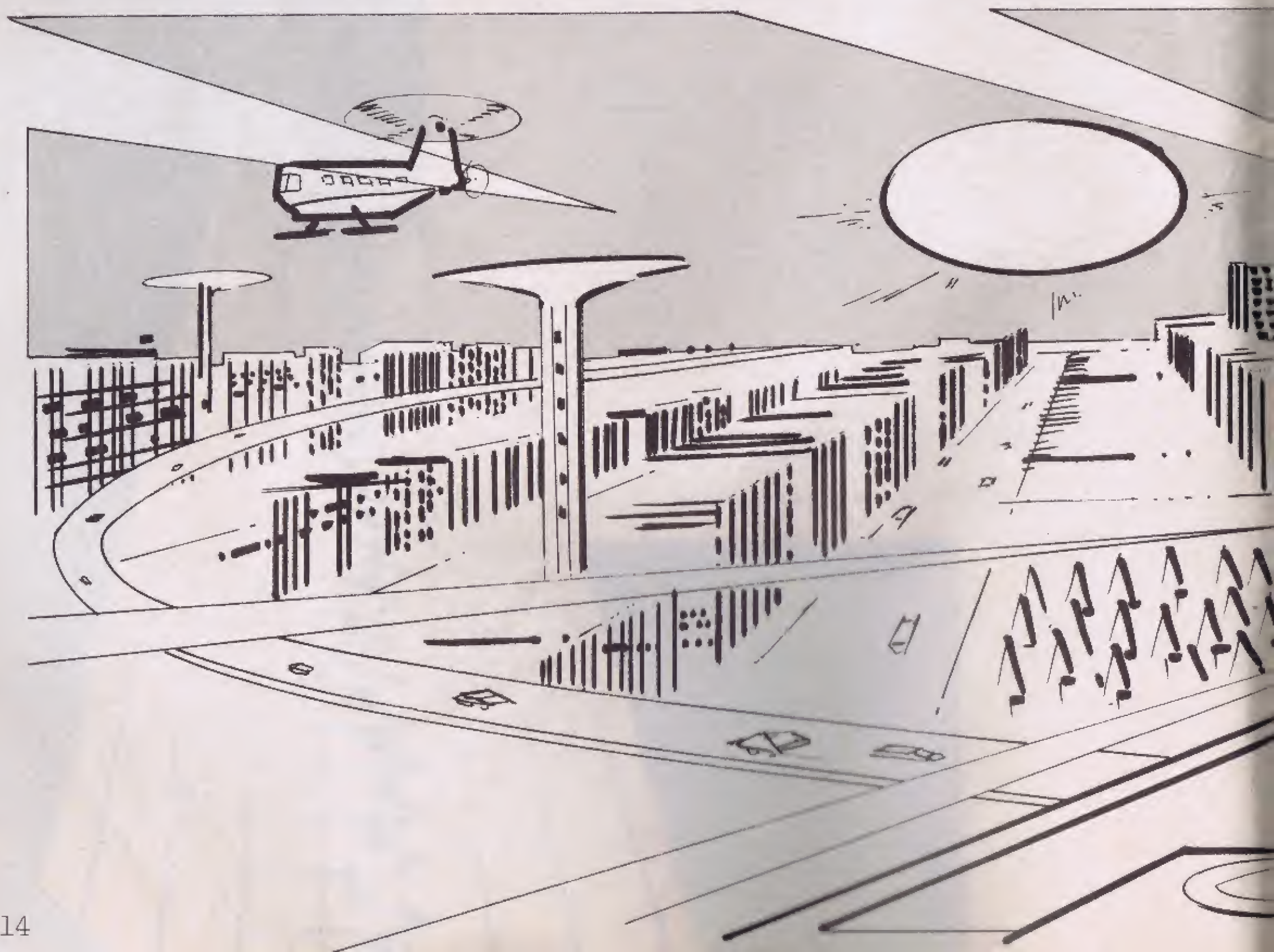
流れ図Bは、テープからの読み込みのためにファイルをOPENした時のコンピュータ・ユーザー間のやりとりとコンピュータの動作がおおまかに示してあります。読み込みファイルのOPENには2通りの場合があります。

(1)テープからのデータ読み込みのためのOPEN
(2)プログラムをメモリにLOADする

キーボードから直接キーインすることによってダイレクトにファイル・オープンした場合、“PRESS PLAY…”“SEARCHING FOR”そして“FOUNDファイル名”が表示されます。LOADを使用した場合には、ロードされたプログラムの変数はすべて初期化されます。

プログラムからファイルがオープンされた場合、すでにPLAYボタンが押してあればメッセージは何も表示されません。つまり、画面の表示内容が壊されてしまうことはありません。また、この場合BASIC変数の初期化はおこなわれません（つまり、オーバレイ時のパラメータ引渡しができます)。

INPUT #, LOAD時における読み込みプロセス

COMMODORE
DETAILED MEMORY MAP

コモドール64の詳細メモリ・マップを初公開！このメモリ・マップもVIC-1001のメモリ・マップ(『VIC！』4号，7号掲載)同様，カナダのジム・バターフィールドの解析によります。コモドール64のシステムRAMメモリ（＄0000～＄0400)はVIC-1001とほぼ同一。一番重要な違いは＄0000と＄0001番地で、コモドール64ではこの番地が6510CPUのI/Oレジスタとして使われていて、USR関数のためのJUMP番地は＄0311，＄0312になっています。
BASIC ROMは$A000～$BFFF, KERNAL ROMは$E000～$FFFFまでです。より詳しくいえば、BASICインタプリタは$A000～$BFFF，$E000～$E4FF，KERNAL(オペレーション・システムとほぼ同義)は$E500～$FFFFを占めています。

VIC! SPECIAL

Commodore-64 Architecture Map

Address	Decimal	Main map	Alternate
$FFFF	(65535)	KERNAL ROM	8K HI RAM (bit 1 of $0001) 0 = RAM 1 = KERNAL ROM
$E000	(57344)	CIA 1, CIA 2 (S. Bus, P.U.P)	4K RAM maps to $D000 when bits 0 & 1 of $0001 = 0
$DC00	(55320)	Colour Ram Nibbles	
$D800	(55296)	VIC II, SID	
$D000	(53248)	4K RAM	
$C000	(49152)	BASIC INTERPRETER ROM	8K LO RAM (bit 0 of $0001) 0 = RAM 1 = BASIC ROM
$A000	(40960)	BASIC User RAM (36864 Bytes)	EXROM 8K ROM Cartridge maps here
$8000	(32768)		
$4000	(16384)		HI-RES Screen maps here
$2000	(8192)		
$1000	(4096)	スプライト／プログラマブル・キャラクタ用リザーブ・エリア	
$0800	(2048)	Screen (1K)	HI-RES Colour Table
$0400	(1024)	Workspace (1K)	
		Processor Reg ($0/1)	
$0000			

CHARACTER SET
(bit 2 of $0001)
0 = CHAR ROM
1 = I/O ROM/RAM

VIC II Chip sees this 16K block on power-up ($0000–$4000)

注：コモドール64USAバージョンには$0800～0FFFのスプライト/プログラマブル·キャラクタ用リザーブ·エリアがなく、BASICテキスト·スタート番地が$0801となっています。JISバージョンをUSAバージョンにあわせるためには、POKE44,8：POKE 8✳256,0：CLR：NEWをおこなってください。

Commodore 64 Memory Map

Compiled by
Jim Butterfield

0000		0	Chip directional register
0001		1	Chip I/O; memory & tape control
0003	-0004	3-4	Float-Fixed vector
0005	-0006	5-6	Fixed-Float vector
0007		7	Search character
0008		8	Scan-quotes flag
0009		9	TAB column save
000A		10	0 = LOAD, 1 = VERIFY
000B		11	Input buffer pointer/# subscrpt
000C		12	Default DIM flag
000D		13	Type: FF = string, 00 = numeric
000E		14	Type: 80 = integer, 00 = floating point
000F		15	DATA scan/LIST quote/memry flag
0010		16	Subscript/FNx flag
0011		17	0 = INPUT;$40 = GET;$98 = READ
0012		18	ATN sign/Comparison eval flag
0013		19	Current I/O prompt flag
0014	-0015	20-21	Integer value
0016		22	Pointer: temporary string stack
0017	-0018	23-24	Last temp string vector
0019	-0021	25-33	Stack for temporary strings
0022	-0025	34-37	Utility pointer area
0026	-002A	38-42	Product area for multiplication
002B	-002C	43-44	Pointer: Start-of-Basic
002D	-002E	45-46	Pointer: Start-of-Variables
002F	-0030	47-48	Pointer: Start-of-Arrays
0031	-0032	49-50	Pointer: End-of-Arrays
0033	-0034	51-52	Pointer: String-storage(moving down)
0035	-0036	53-54	Utility string pointer
0037	-0038	55-56	Pointer: Limit-of-memory
0039	-003A	57-58	Current Basic line number
003B	-003C	59-60	Previous Basic line number
003D	-003E	61-62	Pointer: Basic statement for CONT
003F	-0040	63-64	Current DATA line number
0041	-0042	65-66	Current DATA address
0043	-0044	67-68	Input vector
0045	-0046	69-70	Current variable name
0047	-0048	71-72	Current variable address
0049	-004A	73-74	Variable pointer for FOR/NEXT
004B	-004C	75-76	Y-save; op-save; Basic pointer save
004D		77	Comparison symbol accumulator
004E	-0053	78-83	Misc work area, pointers, etc
0054	-0056	84-86	Jump vector for functions
0057	-0060	87-96	Misc numeric work area
0061		97	Accum#1: Exponent
0062	-0065	98-101	Accum#1: Mantissa
0066		102	Accum#1: Sign
0067		103	Series evaluation constant pointer
0068		104	Accum#1 hi-order (overflow)
0069	-006E	105-110	Accum#2: Exponent, etc.
006F		111	Sign comparison, Acc#1 vs #2

0070		112	Accum#1 lo-order (rounding)
0071	-0072	113-114	Cassette buff len/Series pointer
0073	-008A	115-138	CHRGET subroutine; get Basic char
007A	-007B	122-123	Basic pointer (within subrtn)
008B	-008F	139-143	RND seed value
0090		144	Status word ST
0091		145	Keyswitch PIA: STOP and RVS flags
0092		146	Timing constant for tape
0093		147	Load = 0, Verify = 1
0094		148	Serial output: deferred char flag
0095		149	Serial deferred character
0096		150	Tape EOT received
0097		151	Register save
0098		152	How many open files
0099		153	Input device, normally 0
009A		154	Output CMD device, normally 3
009B		155	Tape character parity
009C		156	Byte-received flag
009D		157	Direct = $80/RUN = 0 output control
009E		158	Tp Pass 1 error log/char buffer
009F		159	Tp Pass 2 err log corrected
00A0	-00A2	160-162	Jiffy Clock HML
00A3		163	Serial bit count/EOI flag
00A4		164	Cycle count
00A5		165	Countdown,tape write/bit count
00A6		166	Tape buffer pointer
00A7		167	Tp Wrt ldr count/Rd pass/inbit
00A8		168	Tp Wrt new byte/Rd error/inbit cnt
00A9		169	Wrt start bit/Rd bit err/stbit
00AA		170	Tp Scan;Cnt;Ld;End/byte assy
00AB		171	Wr lead length/Rd checksum/parity
00AC	-00AD	172-173	Pointer: tape bufr, scrolling
00AE	-00AF	174-175	Tape end adds/End of program
00B0	-00B1	176-177	Tape timing constants
00B2	-00B3	178-179	Pntr: start of tape buffer
00B4		180	1 = Tp timer enabled; bit count
00B5		181	Tp EOT/RS232 next bit to send
00B6		182	Read character error/outbyte buf
00B7		183	# characters in file name
00B8		184	Current logical file
00B9		185	Current secndy address
00BA		186	Current device
00BB	-00BC	187-188	Pointer to file name
00BD		189	Wr shift word/Rd input char
00BE		190	# blocks remaining to Wr/Rd
00BF		191	Serial word buffer
00C0		192	Tape motor interlock
00C1	-00C2	193-194	I/O start address
00C3	-00C4	195-196	Kernel setup pointer
00C5		197	Last key pressed
00C6		198	# chars in keybd buffer
00C7		199	Screen reverse flag
00C8		200	End-of-line for input pointer
00C9	-00CA	201-202	Input cursor log (row, column)
00CB		203	Which key: 64 if no key

00CC		204	0 = flash cursor
00CD		205	Cursor timing countdown
00CE		206	Character under cursor
00CF		207	Cursor in blink phase
00D0		208	Input from screen/from keyboard
00D1	-00D2	209-210	Pointer to screen line
00D3		211	Position of cursor on above line
00D4		212	0 = direct cursor, else programmed
00D5		213	Current screen line length
00D6		214	Row where curosr lives
00D7		215	Last inkey/checksum/buffer
00D8		216	# of INSERTs outstanding
00D9	-00F2	217-242	Screen line link table
00F3	-00F4	243-244	Screen color pointer
00F5	-00F6	245-246	Keyboard pointer
00F7	-00F8	247-248	RS-232 Rcv pntr
00F9	-00FA	249-250	RS-232 Tx pntr
00FF	-010A	256-266	Floating to ASCII work area
0100	-013E	256-318	Tape error log
0100	-01FF	256-511	Processor stack area
0200	-0258	512-600	Basic input buffer
0259	-0262	601-610	Logical file table
0263	-026C	611-620	Device # table
026D	-0276	621-630	Sec Adds table
0277	-0280	631-640	Keybd buffer
0281	-0282	641-642	Start of Basic Memory
0283	-0284	643-644	Top of Basic Memory
0285		645	Serial bus timeout flag
0286		646	Current color code
0287		647	Color under cursor
0288		648	Screen memory page
0289		649	Max size of keybd buffer
028A		650	Repeat all keys
028B		651	Repeat speed counter
028C		652	Repeat delay counter
028D		653	Keyboard Shift/Control flag
028E		654	Last shift pattern
028F	-0290	655-656	Keyboard table setup pointer
0291		657	Keyboard shift mode
0292		658	0 = scroll enable
0293		659	RS-232 control reg
0294		660	RS-232 command reg
0295	-0296	661-662	Bit timing
0297		663	RS-232 status
0298		664	# bits to send
0299	-029A	665	RS-232 speed/code
029B		667	RS232 receive pointer
029C		668	RS232 input pointer
029D		669	RS232 transmit pointer
029E		670	RS232 output pointer
029F	-02A0	671-672	IRQ save during tape I/O
02A1		673	CIA 2 (NMI) Interrupt Control
02A2		674	CIA 1 Timer A control log
02A3		675	CIA 1 Interrupt Log
02A4		676	CIA 1 Timer A enabled flag

Hex	Decimal	Description	
02A5	677	Screen row marker	
02C0 -02FE	704–766	(Sprite 11)	
0300 -0301	768–769	Error message link	
0302 -0303	770–771	Basic warm start link	
0304 -0305	772–773	Crunch Basic tokens link	
0306 -0307	774–775	Print tokens link	
0308 -0309	776–777	Start new Basic code link	
030A -030B	778–779	Get arithmetic element link	
030C	780	SYS A–reg save	
030D	781	SYS X–reg save	
030E	782	SYS Y–reg save	
030F	783	SYS status reg save	
0310 -0312	784–785	USR function jump	(B248)
0314 -0315	788–789	Hardware interrupt vector	(EA31)
0316 -0317	790–791	Break interrupt vector	(FE66)
0318 -0319	792–793	NMI interrupt vector	(FE47)
031A -031B	794–795	OPEN vector	(F34A)
031C -031D	796–797	CLOSE vector	(F291)
031E -031F	798–799	Set–input vector	(F20E)
0320 -0321	800–801	Set–output vector	(F250)
0322 -0323	802–803	Restore I/O vector	(F333)
0324 -0325	804–805	INPUT vector	(F157)
0326 -0327	806–807	Output vector	(F1CA)
0328 -0329	808–809	Test–STOP vector	(F6ED)
032A -032B	810–811	GET vector	(F13E)
032C -032D	812–813	Abort I/O vector	(F32F)
032E -032F	814–815	Warm start vector	(FE66)
0330 -0331	816–817	LOAD link	(F4A5)
0332 -0333	818–819	SAVE link	(F5ED)
033C -03FB	828–1019	Cassette buffer	
0340 -037E	832–894	(Sprite 13)	
0380 -03BE	896–958	(Sprite 14)	
03C0 -03FE	960–1022	(Sprite 15)	
0400 -07FF	1024–2047	Screen memory	
0800 -9FFF	2048–40959	Basic RAM memory	
8000 -9FFF	32768–40959	Alternate: ROM plug–in area	
A000 -BFFF	40960–49151	ROM: Basic	
A000 -BFFF	49060–59151	Alternate: RAM	
C000 -CFFF	49152–53247	RAM memory, including alternate	
D000 -D02E	53248–53294	Video Chip (6566)	
D400 -D41C	54272–54300	Sound Chip (6581 SID)	
D800 -DBFF	55296–56319	Color nybble memory	
DC00 -DC0F	56320–56335	Interface chip 1, IRQ (6526 CIA)	
DD00 -DD0F	56576–56591	Interface chip 2, NMI (6526 CIA)	
D000 -DFFF	53248–53294	Alternate: Character set	
E000 -FFFF	57344–65535	ROM: Operating System	
E000 -FFFF	57344–65535	Alternate: RAM	
FF81 -FFF5	65409–65525	Jump Table, Including:	
FFC6		– Set Input channel	
FFC9		– Set Output channel	
FFCC		– Restore default I/O channels	
FFCF		– INPUT	
FFD2		– PRINT	
FFE1		– Test Stop key	
FFE4		– GET	

Commodore 64 – ROM Memory Map

A000; ROM control vectors
A00C; Keyword action vectors
A052; Function vectors
A080; Operator vectors
A09E; Keywords
A19E; Error messages
A328; Error message vectors
A365; Misc messages
A38A; Scan stack for FOR/GOSUB
A3B8; Move memory
A3FB; Check stack depth
A408; Check memory space
A435; 'out of memory'
A437; Error routine
A469; BREAK entry
A474; 'ready.'
A480; Ready for Basic
A49C; Handle new line
A533; Re–chain lines
A560; Receive input line
A579; Crunch tokens
A613; Find Basic line
A642; Perform [NEW]
A65E; Perform [CLR]
A68E; Back up text pointer
A69C; Perform [LIST]
A742; Perform [FOR]
A7ED; Execute statement
A81D; Perform [RESTORE]
A82C; Break
A82F; Perform [STOP]
A831; Perform [END]
A857; Perform [CONT]
A871; Perform [RUN]
A883; Perform [GOSUB]
A8A0; Perform [GOTO]
A8D2; Perform [RETURN]
A8F8; Perform [DATA]
A906; Scan for next statement
A928; Perform [IF]
A93B; Perform [REM]
A94B; Perform [ON]
A96B; Get fixed point number
A9A5; Perform [LET]
AA80; Perform [PRINT#]
AA86; Perform [CMD]
AAA0; Perform [PRINT]
AB1E; Print string from (y.a)
AB3B; Print format character
AB4D; Bad input routine
AB7B; Perform [GET]
ABA5; Perform [INPUT#]
ABBF; Perform [INPUT]
ABF9; Prompt & input
AC06; Perform [READ]
ACFC; Input error messages
AD1E; Perform [NEXT]
AD78; Type match check
AD9E; Evaluate expression
AEA8; Constant – pi
AEF1; Evaluate within brackets
AEF7; ')'
AEFF; comma..
AF08; Syntax error
AF14; Check range
AF28; Search for variable
AFA7; Setup FN reference
AFE6; Perform [OR]
AFE9; Perform [AND]
B016; Compare
B081; Perform [DIM]
B08B; Locate variable
B113; Check alphabetic
B11D; Create variable
B194; Array pointer subrtine
B1A5; Value 32768
B1B2; Float–fixed
B1D1; Set up array
B245; 'bad subscript'
B248; 'illegal quantity'
B34C; Compute array size
B37D; Perform [FRE]
B391; Fix–float
B39E; Perform [POS]
B3A6; Check direct
B3B3; Perform [DEF]
B3E1; Check fn syntax
B3F4; Perform [FN]
B465; Perform [STR$]
B475; Calculate string vector
B487; Set up string
B4F4; Make room for string
B526; Garbage collection
B5BD; Check salvageability
B606; Collect string
B63D; Concatenate
B67A; Build string to memory
B6A3; Discard unwanted string
B6DB; Clean descriptor stack
B6EC; Perform [CHR$]
B700; Perform [LEFT$]
B72C; Perform [RIGHT$]
B737; Perform [MID$]
B761; Pull string parameters
B77C; Perform [LEN]
B782; Exit string–mode
B78B; Perform [ASC]
B79B; Input byte paramter
B7AD; Perform [VAL]
B7EB; Parameters for POKE/WAIT
B7F7; Float–fixed
B80D; Perform [PEEK]
B824; Perform [POKE]
B82D; Perform [WAIT]

Address	Routine
B849;	Add 0.5
B850;	Subtract-from
B853;	Perform [subtract]
B86A;	Perform [add]
B947;	Complement FAC#1
B97E;	'overflow'
B983;	Multiply by zero byte
B9EA;	Perform [LOG]
BA2B;	Perform [multiply]
BA59;	Multiply-a-bit
BA8C;	Memory to FAC#2
BAB7;	Adjust FAC#1/#2
BAD4;	Underflow/overflow
BAE2;	Multiply by 10
BAF9;	+ 10 in floating pt
BAFE;	Divide by 10
BB12;	Perform [divide]
BBA2;	Memory to FAC#1
BBC7;	FAC#1 to memory
BBFC;	FAC#2 to FAC#1
BC0C;	FAC#1 to FAC#2
BC1B;	Round FAC#1
BC2B;	Get sign
BC39;	Perform [SGN]
BC58;	Perform [ABS]
BC5B;	Compare FAC#1 to mem
BC9B;	Float-fixed
BCCC;	Perform [int]
BCF3;	String to FAC
BD7E;	Get ascii digit
BDC2;	Print 'IN..'
BDCD;	Print line number
BDDD;	Float to ascii
BF16;	Decimal constants
BF3A;	TI constants
BF71;	Perform [SQR]
BF7B;	Perform [power]
BFB4;	Perform [negative]
BFED;	Perform [EXP]
E043;	Series eval 1
E059;	Series eval 2
E097;	Perform [RND]
E0f9;	?? breakpoints ??
E12A;	Perform [SYS]
E156;	Perform [SAVE]
E165;	Perform [VERIFY]
E168;	Perform [LOAD]
E1BE;	Perform [OPEN]
E1C7;	Perform [CLOSE]
E1D4;	Parameters for LOAD/SAVE
E206;	Check default parameters
E20E;	Check for comma
E219;	Parameters for open/close
E264;	Perform [COS]
E26B;	Perform [SIN]
E2B4;	Perform [TAN]
E30E;	Perform [ATN]
E37B;	Warm restart
E394;	Initialize
E3A2;	CHRGET for zero page
E3BF;	Initialize Basic
E447;	Vectors for $300
E453;	Initialize vectors
E45F;	Power-up message
E500;	Get I/O address
E505;	Get screen size
E50A;	Put/get row/column
E518;	InitializeI/O
E544;	Clear screen
E566;	Home cursor
E56C;	Set screen pointers
E5A0;	Set I/O defaults
E5B4;	Input from keyboard
E632;	Input from screen
E684;	Quote test
E691;	Setup screen print
E6B6;	Advance cursor
E6ED;	Retreat cursor
E701;	Back into previous line
E716;	Output to screen
E87C;	Go to next line
E891;	Perform <return>
E8A1;	Check line decrement
E8B3;	Check line increment
E8CB;	Set color code
E8DA;	Color code table
E8EA;	Scroll screen
E965;	Open space on screen
E9C8;	Move a screen line
E9E0;	Synchronize color transfer
E9F0;	Set start-of-line
E9FF;	Clear screen line
EA13;	Print to screen
EA24;	Synchronize color pointer
EA31;	Interrupt – clock etc
EA87;	Read keyboard
EB79;	Keyboard select vectors
EB81;	Keyboard 1 – unshifted
EBC2;	Keyboard 2 – shifted
EC03;	Keyboard 3 – 'comm'
EC44;	Graphics/text contrl
EC4F;	Set graphics/text mode
EC78;	Keyboard 4
ECB9;	Video chip setup
ECE7;	Shift/run equivalent
ECF0;	Screen ln address low
ED09;	Send 'talk'
ED0C;	Send 'listen'
ED40;	Send to serial bus
EDB2;	Serial timeout
EDB9;	Send listen SA
EDBE;	Clear ATN
EDC7;	Send talk SA
EDCC;	Wait for clock
EDDD;	Send serial deferred
EDEF;	Send 'untalk'

EDFE;	Send 'unlisten'
EE13;	Receive from serial bus
EE85;	Serial clock on
EE8E;	Serial clock off
EE97;	Serial output '1'
EEA0;	Serial output '0'
EEA9;	Get serial in & clock
EEB3;	Delay 1 ms
EEBB;	RS-232 send
EF06;	Send new RS-232 byte
EF2E;	No-DSR error
EF31;	No-CTS error
EF3B;	Disable timer
EF4A;	Compute bit count
EF59;	RS232 receive
EF7E;	Setup to receive
EFC5;	Receive parity error
EFCA;	Receive overflow
EFCD;	Receive break
EFD0;	Framing error
EFE1;	Submit to RS232
F00D;	No-DSR error
F017;	Send to RS232 buffer
F04D;	Input from RS232
F086;	Get from RS232
F0A4;	Check serial bus idle
F0BD;	Messages
F12B;	Print if direct
F13E;	Get..
F14E;	..from RS232
F157;	Input
F199;	Get.. tape/serial/rs232
F1CA;	Output..
F1DD;	..to tape
F20E;	Set input device
F250;	Set output device
F291;	Close file
F30F;	Find file
F31F;	Set file values
F32F;	Abort all files
F333;	Restore default I/O
F34A;	Do file open
F3D5;	Send SA
F409;	Open RS232
F49E;	Load program
F5AF;	'searching'
F5C1;	Print filename
F5D2;	'loading/verifying'
F5DD;	Save program
F68F;	Print 'saving'
F69B;	Bump clock
F6BC;	Log PIA key reading
F6DD;	Get time
F6E4;	Set time
F6ED;	Check stop key
F6FB;	Output error messages
F72D;	Find any tape headr
F76A;	Write tape header
F7D0;	Get buffer address
F7D7;	Set buffer start/end pointers
F7EA;	Find specific header
F80D;	Bump tape pointer
F817;	'press play..'
F82E;	Check tape status
F838;	'press record..'
F841;	Initiate tape read
F864;	Initiate tape write
F875;	Common tape code
F8D0;	Check tape stop
F8E2;	Set read timing
F92C;	Read tape bits
FA60;	Store tape chars
FB8E;	Reset pointer
FB97;	New character setup
FBA6;	Send transition to tape
FBC8;	Write data to tape
FBCD;	IRQ entry point
FC57;	Write tape leader
FC93;	Restore normal IRQ
FCB8;	Set IRQ vector
FCCA;	Kill tape motor
FCD1;	Check r/w pointer
FCDB;	Bump r/w pointer
FCE2;	Power reset entry
FD02;	Check 8-rom
FD10;	8-rom mask
FD15;	Kernal reset
FD1A;	Kernal move
FD30;	Vectors
FD50;	Initialize system constnts
FD9B;	IRQ vectors
FDA3;	Initialize I/O
FDDD;	Enable timer
FDF9;	Save filename data
FE00;	Save file details
FE07;	Get status
FE18;	Flag status
FE1C;	Set status
FE21;	Set timeout
FE25;	Read/set top of memory
FE27;	Read top of memory
FE2D;	Set top of memory
FE34;	Read/set bottom of memory
FE43;	NMI entry
FE66;	Warm start
FEB6;	Reset IRQ & exit
FEBC;	Interrupt exit
FEC2;	RS-232 timing table
FED6;	NMI RS-232 in
FF07;	NMI RS-232 out
FF43;	Fake IRQ
FF48;	IRQ entry
FF81;	Jumbo jump table
FFFA;	Hardware vectors

Processor I/O Port (6510)

$0000	IN	IN	OUT	IN	OUT	OUT	OUT	OUT	DDR 0
$0001			Tape Motor	Tape Sense	Tape Write	D-ROM Switch	EF RAM Switch	AB RAM Switch	PR 1

SID (6581)

Voice 1	Voice 2	Voice 3	Register	Voice 1	Voice 2	Voice 3
$D400	$D407	$D40E	Frequency L	54272	54279	54286
$D401	$D408	$D40F	Frequency H	54273	54280	54287
$D402	$D409	$D410	Pulse Width L	54274	54281	54288
$D403	$D40A	$D411	0 0 0 0 / Pulse Width H	54275	54282	54289
$D404	$D40B	$D412	Voice Type: NSE, PUL, SAW, TRI / Key	54276	54283	54290
$D405	$D40C	$D413	Attack Time 2ms – 8ms / Decay Time 6ms – 24 sec	54277	54284	54291
$D406	$D40D	$D414	Sustain Level / Release Time 6ms 24 sec	54278	54285	54292

Voices (write only)

Address	Register	Decimal
$D415	0 0 0 0 0 / Filter Frequency L	54293
$D416	Filter Frequency H	54294
$D417	Resonance / Filter Voices: Ext, V3, V2, V1	54295
$D418	V3 off / Passband: HI, BP, LO / Master Volume	54296

Filter & Volume (write only)

Address	Register	Decimal
$D419	Paddle X (A/D #1)	54297
$D41A	Paddle Y (A/D #2)	54298
$D41B	Noise 3 (random)	54299
$D41C	Envelope 3	54300

Sense (read only)

Note: Special Voice Features
(TEST, RING MOD, SYNC)
are omitted from the above diagram.

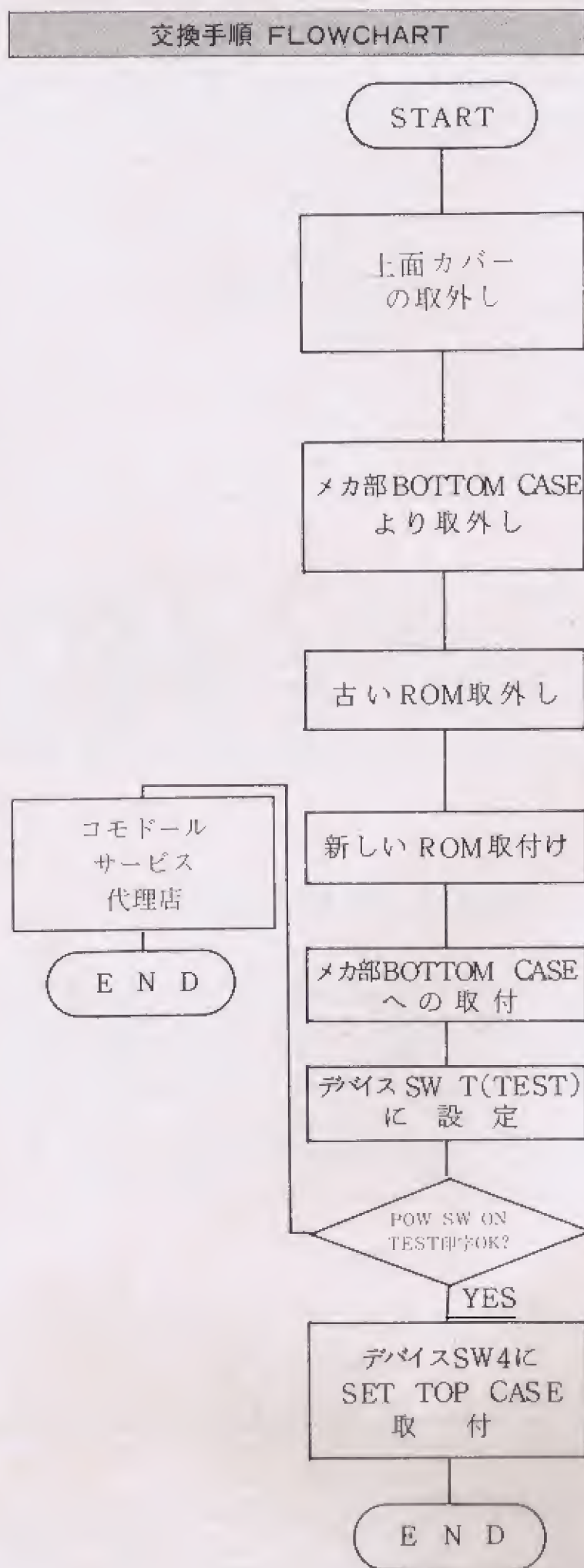
CIA 1 (IRQ) (6526)

CIA 2 (NMI) (6526)

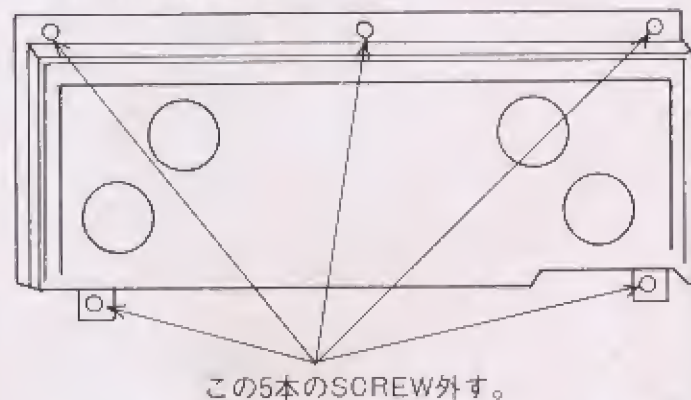
* Connected but not used by O.S.

VIC-1515・15

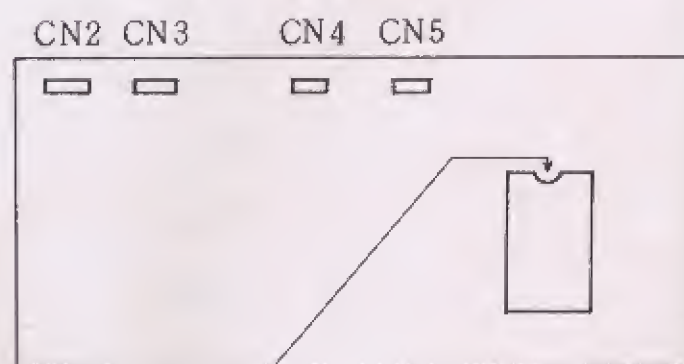
買っちゃった！COMMODORE64

そこで、VIC-1001で使ってたプリンタとフロッピーを交換しましょ。手順はカンタン、僕でもラクラクだよ。

1) 上面カバーの取外し方法
- VIC-1515の場合
  裏面より2本、SCREW2本SCREWを外すと上面カバーが外れます。
- VIC-1525の場合
  裏面より5本のSCREWを外すと上面カバーが外れます。

〈VIC-1525裏面図〉

この5本のSCREW外す。

2) メカ部BOTTOM CASEよりの取外し方法
(VIC-1515, 1525共に同じ)

3) 古いROMの取外し方法
基盤ユニット見取図参照。
ソケットとROMの間隙にマイナスドライバーを入れ、テコ原理を利用して下さい。

4) 新しいROMの取付方法
古いROMを取外したP4に取付ける。
この場合ROMの向きが有りますので注意して下さい。

この切込が上に来る様にして下さい。

25のROM交換手順書

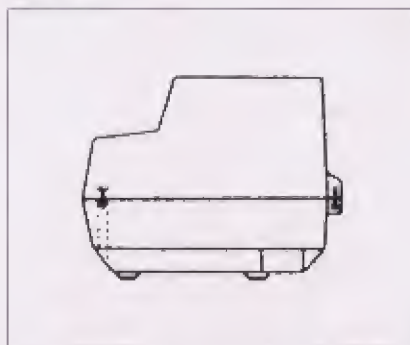
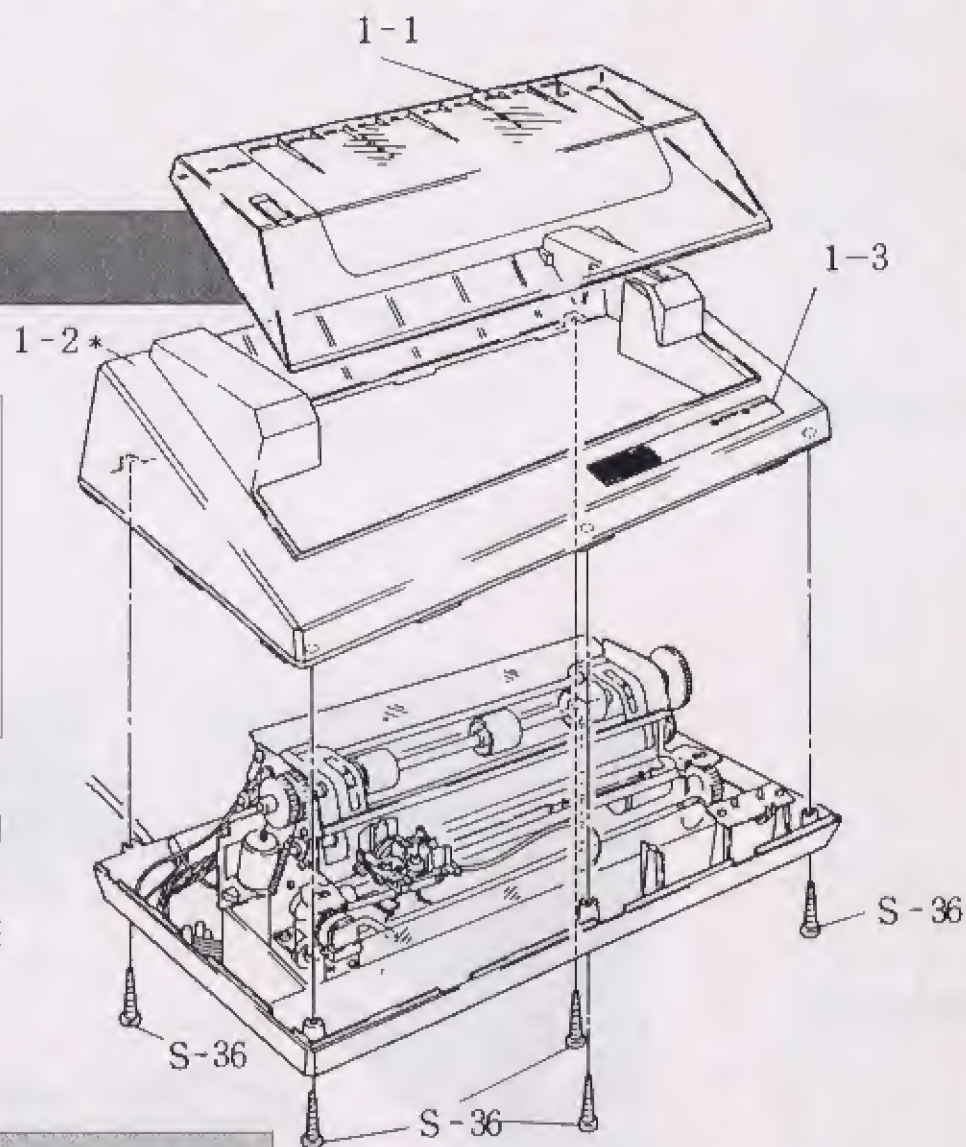
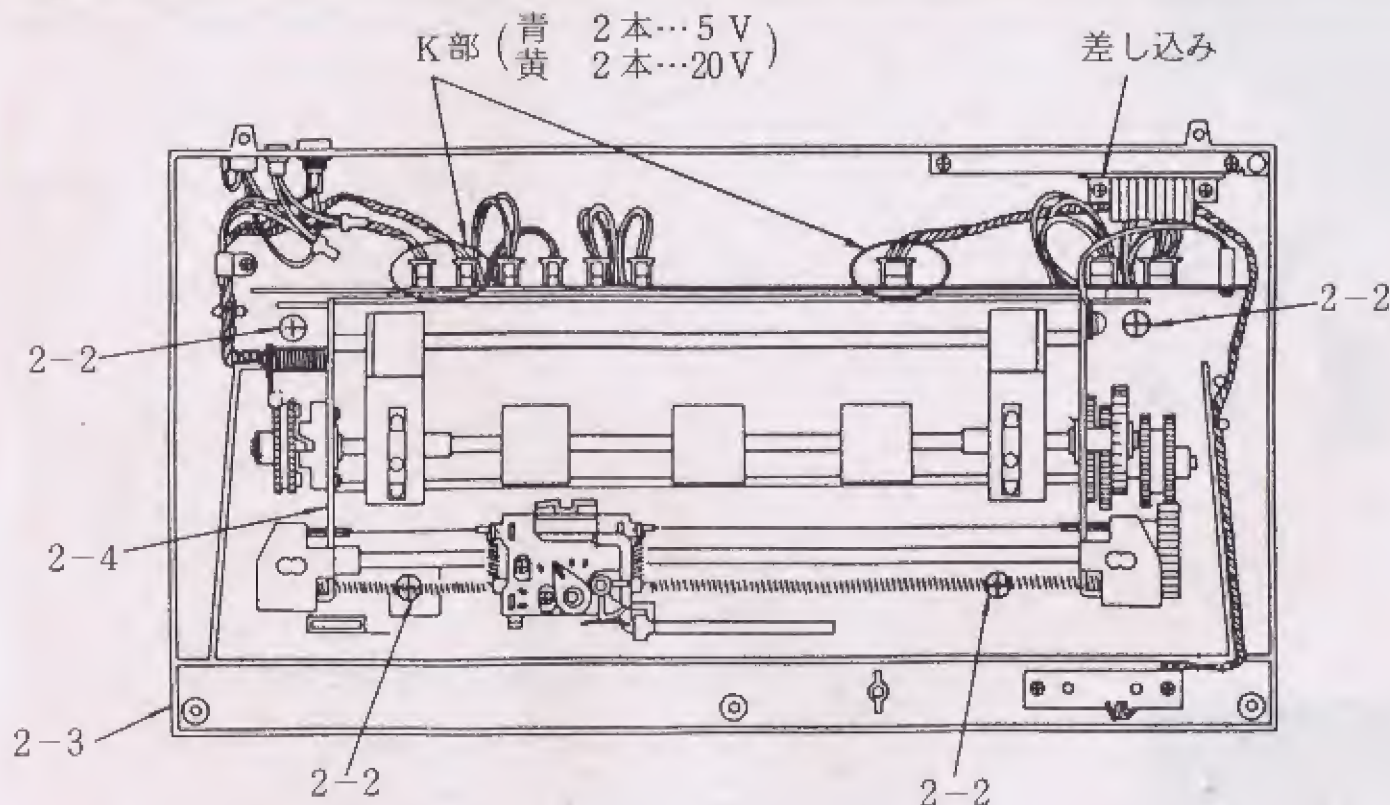
分解・組立

● 分解・組立は各図ごとの部品名記載順、逆記載順に行います。

1. 上枠関係

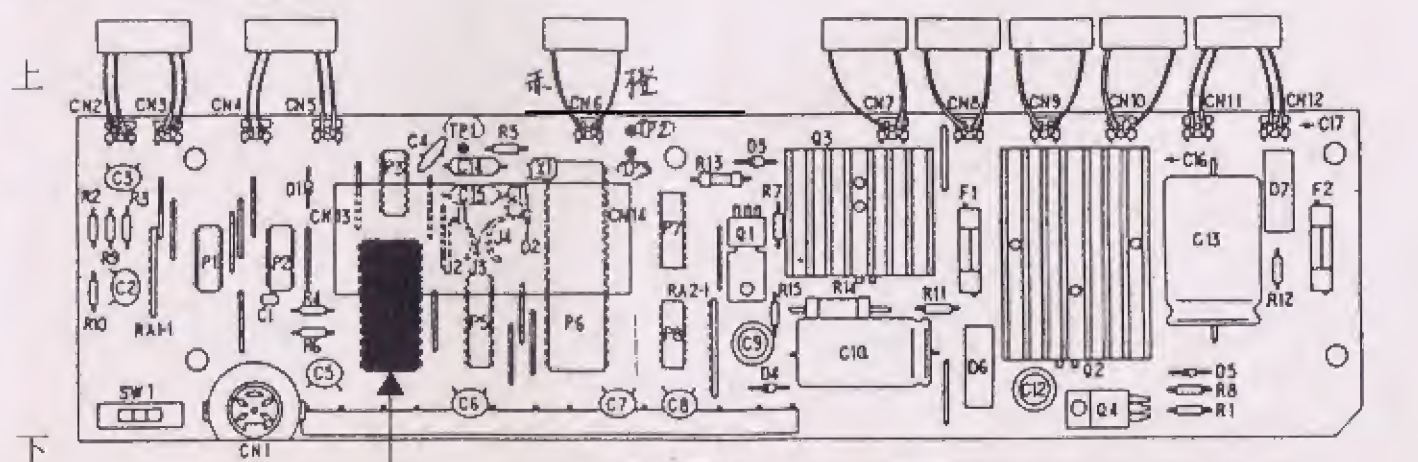
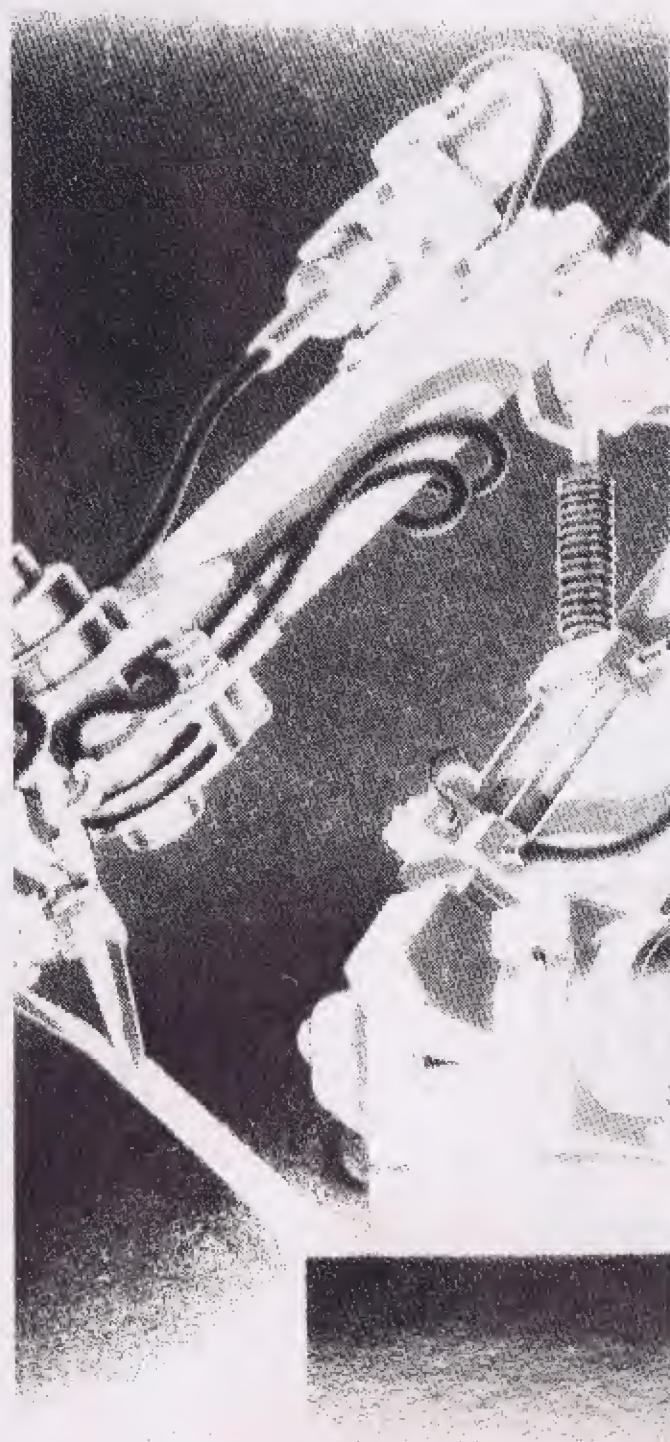
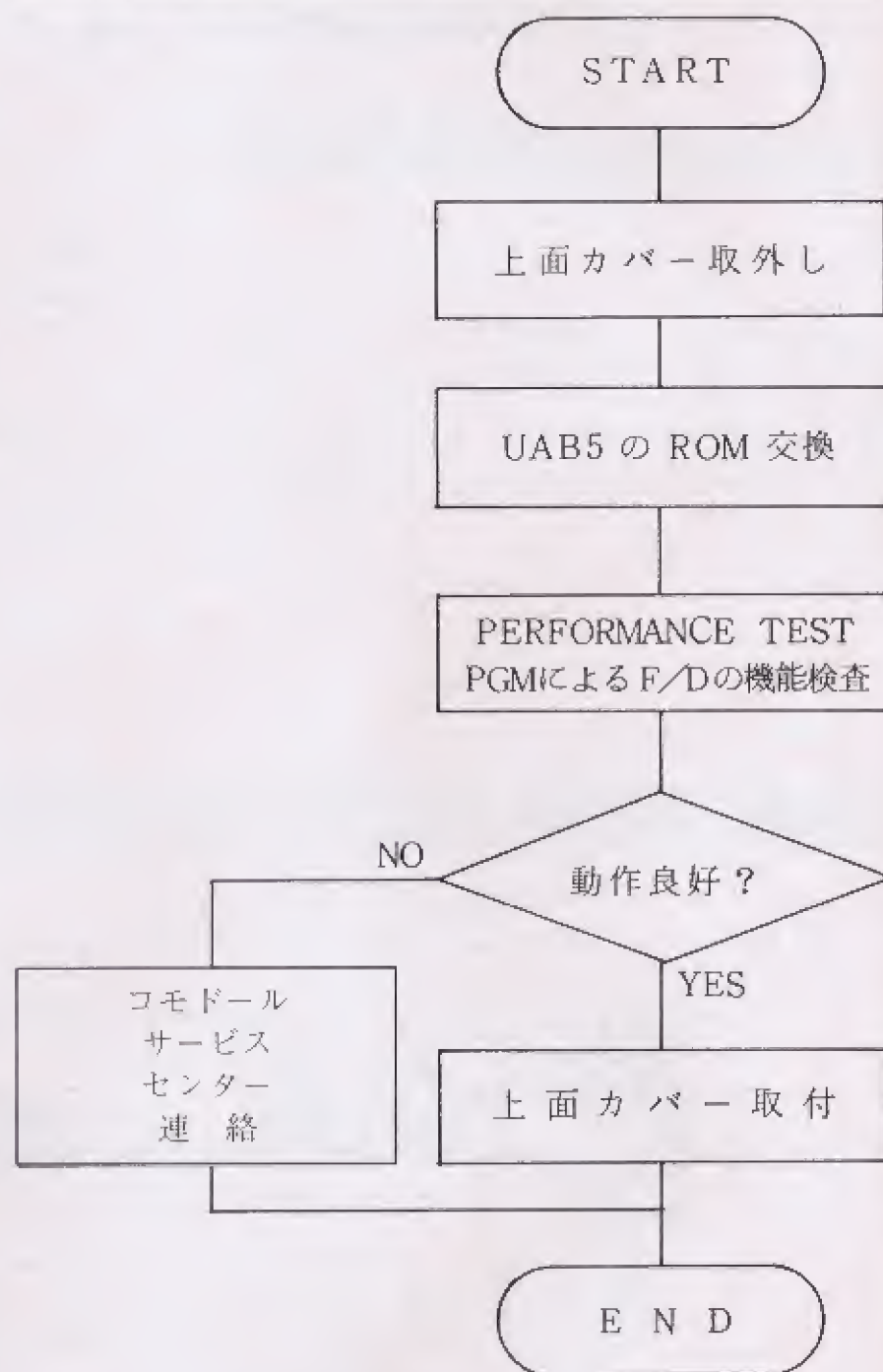
1－1……プリンタカバー
S－36……ネ　　ジ
1－2……上　　枠
1－3……製 品 銘 板

＊上枠を取り付けるときは、パワーランプやエラーランプが穴からでるように注意して下さい。

X視

※1－3を取付ける時、インジケータとランプの関係位置を、ランプ基板ユニットの取付位置を調整してだす。

2. 下枠ブロックと機械体

※1. 2-2により2-3と2-4を着脱する時は、キャリアユニットを中央付近に移動させておくこと。2-3、2-4を結合する時、リード線および復帰バネを2-3と2-4の間にはさみ込んだり痛めたりしないように注意すること。

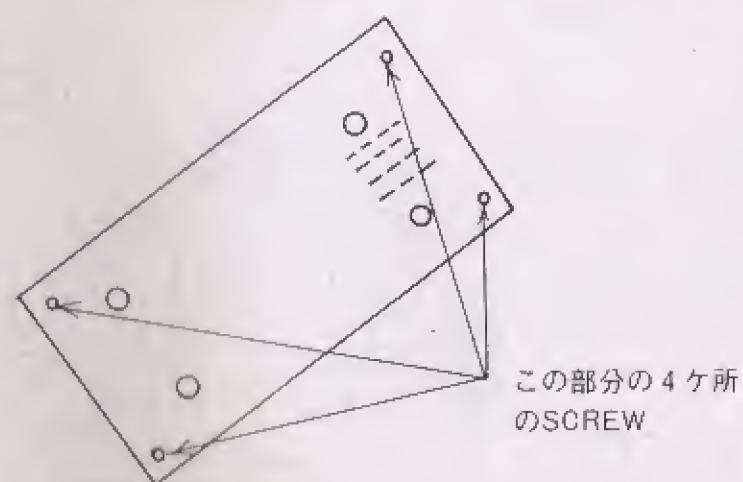
※2. フレキシブルケーブルを痛めないように注意すること。

基板ユニット見取図

VIC-1540→VI

1）交換作業 FLOW CHART

2）上面カバー取外し方法

BOTTOM CASEより4本のSCREWを取外する上面カバーが外れます。

BOTTOM CASE 図

C－1541の変更作業書

3）ROM交換方法

別紙実装図のUAB5の位置のROMをSOCKETとROMの間隙にマイナスドライバーを入れテコの原理を利用して取外す。次に新しいROMを同じ位置に取付ける。

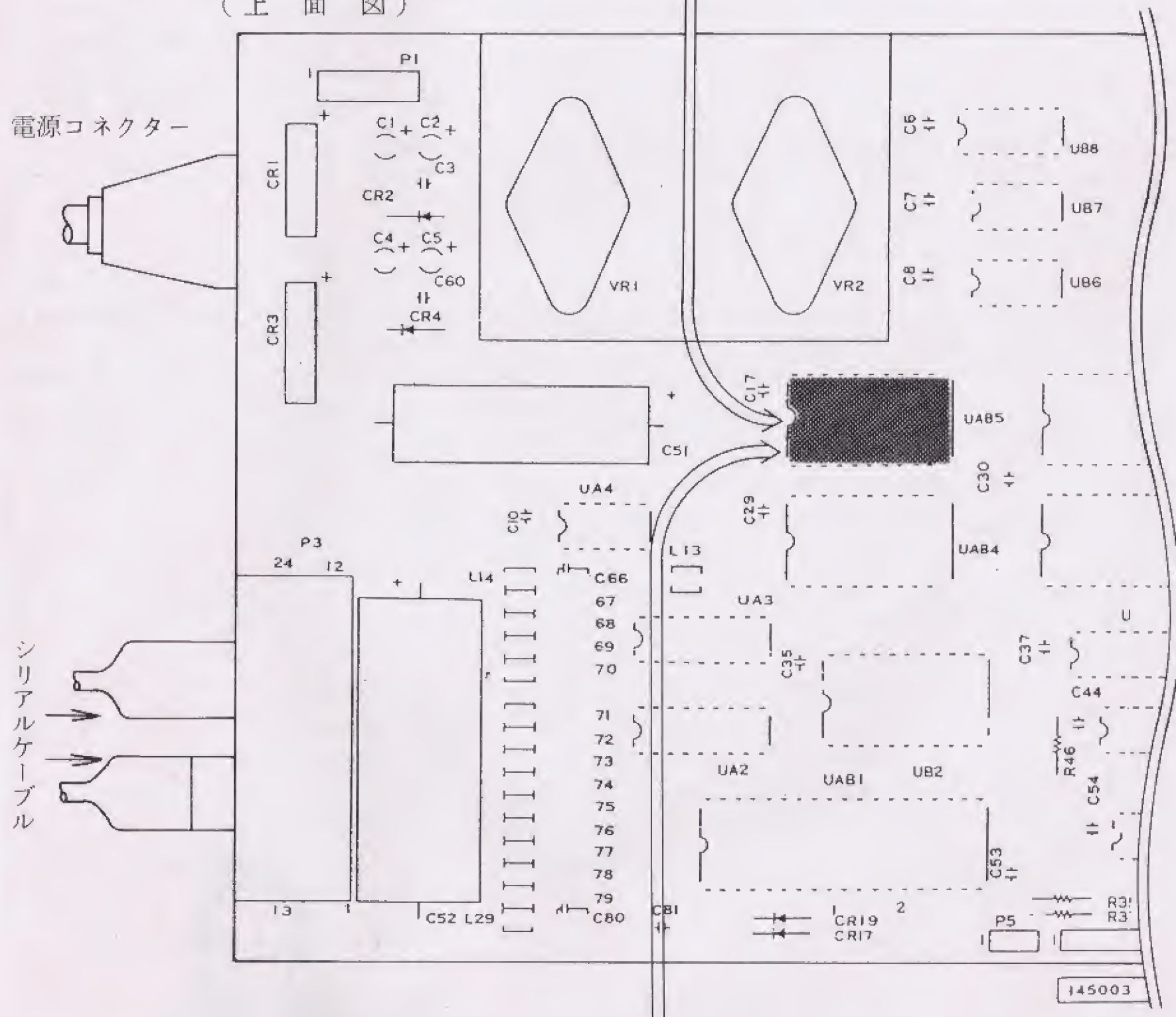
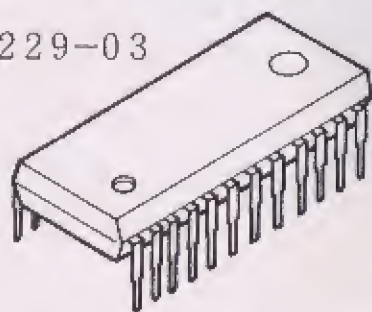
古　い……ROMの部品　No. 325303-01
新しい……ROMの部品　No. 901229-03

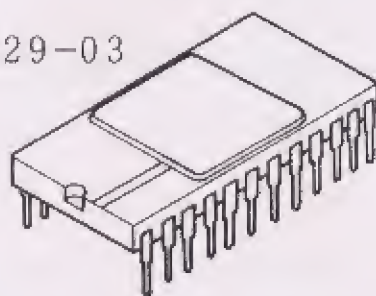
交換されるROM325303-01

（上　面　図）

この図の切り込に合わせてセットする。

電源コネクター

シリアルケーブル

交換時の注意

901229-03

901229-03

上記の図のように取り付ける。

VIC-1001を初期に購入された方で新しく1541を購入される方は、㈱ムーミン・テクニカルセンターまでご相談下さい。
TEL (03) 455-6161

C-64

理美ちゃんがコモドール-64にチャレンジ!!

パソコンデス・マッチ

プリンタって いいのよォ〜。

アクエリアス・カレンダーを作ってみよう

レイモンド 理美

元気でパソコン・フリークしてますかあ？理美は、プリンタを手に入れたので、今まで作ったプログラムや、今作ってるプログラムのリストを、じころ・じころど打ち出して喜んでます。プリンタってあるとないとじゃ大違いなのよね！持ってない頃は長あ〜いプログラム作った時なんか、画面をじと一っとながめて、あわわわになりそうな頭を必死に整理して頑張ってたのよ。

それが今じゃどどっと打ち出して、ぴっぴってカラーペンで直して、ちょこまかっとキーボードたたいて、はい、おしまい。

ムダな時間を使わない合理性、これこそ現代人の命ですね。あはっ。

さんざんリスト出して遊んだあと、リストだけじゃかわいそーだなって思い始めたの。なんか、プリンタさんの役に立つよーなことないかなって悩んだわけ。それで、ビギナーなら必ず1度は通る道、“カレンダー”を作ってみようって決心したんです。それも、今年のだけじゃつまんないから、いつのでも出せるのにしようってー。アイデアはばっつぐんに良かったんだけど、いざ作ってみるとこれがえらくうとましっ子で、非常にたいへんでした。頭がぴこぴこになってしまったわ。

ビギナーの皆さんに告ぐ。

まず、何も考えないでリスト通りに打ち込んでね。そんなに長いプログラムじゃないから大丈夫だと思うけど、ごちゃまちゃした計算のとこや、文字列関数がいりくんでるとこ、ＤＡＴＡ文やなんかはよ〜く気をつけて間違えないように打ってくださいな。打ち終わったらＲＵＮします。年と月を尋ねるメッセージが画面に出てくるから、カレンダーを出したい年月をインプットします。たとえば、1983年10月なら、 1983, 10 っていうふうにね。す、すると、な、なんと、プリンタがじこじこって動いてちゃんとカレンダーを打ち出してくれます。このプログラムはいつのでも出せる（万年）ように作ってあるから、2001年の１月だって、1999年の７の月だって自由自在なのよ。わたしのBirthdayは何曜日だったかな、なんていうのだってばっちりわかっちゃうんだから。好きな時のを打ち出させて、壁にでもはってくださいな。

シニアの皆さんへ、理美のお願い！

とにかく日付の計算がめちゃ難しかったのよ。うるう年の計算はまずぬわんとかなったけど、問題は月による日数の違いと、なんてったって曜日の計算！　月ごとの日数は、DATA文に全部持っちゃうなんてゆー、いもづる簡単方式を採用しちゃって、頭の中味が知れるとこだけど、もっといいアルゴリズムがわかってる人、教えて　　。それから曜日の計算はいっちゃん苦労したんだから、教えてあげないっ。かしこいシニアの皆さんなら、理美が丸１日かかってひねりだしたことでも、すぐ解析しちゃうでしよ。とにかく、そうやってその月の１日の曜日を求めて、あとはカレンダーの形になるようにちょっとくふうしただけなの。

残念なのは、プログラムを短く簡単にしたために、１カ月分のカレンダーしか出せなかったこと。誰か、横３カ月、たて４カ月くらいにして、ストックフォーム１枚に１年分がおさまるようなプログラム作って！

理美のお願い！

いっしょうけんめい待ってるから、VIC情報誌の理美あてに送ってください。

よろしくね。

プリンタ持ってなくても大丈夫よ。

プリンタ持ってないからってそんなに人生をはかなまなくたって大丈夫よ。たった１カ所変えるだけで、ちゃ〜んと画面に出せるんだから。行510に

OPEN 4, 4

っていうステートメントがあるでしょ。それを、

OPEN 4, 3

に変えてくださいな。４を３に変えるだけなのよ。ＲＵＮすると……ほ〜ら画面に出たでしよ。シニアの人ならわかると思うけど、これはね、プリンタの方へ出しなさいって言ってたのを、画面の方にしなさいって変えてあげたのよ。（もっと詳しくわけを知りたい人は“デバイス・ナンバー”っていうものを、研究してくださいな。）

おしまい。

今日の理美のおしゃべりはこれでおしまい。またなんか作ったら　いそいそとみんなにお話してあげるから待っててね。

んじゃ　ばい、ばい。　またねー。

```
100 REM  * CALENDER *
110 PRINT "□CALENDER PRINT" : PRINT
130 INPUT " YEAR,MONTH";Y,M
140 IF Y<=0 THEN 130
150 IF M<=0 OR M>12 THEN 130
200 :
210 FOR I=1 TO M : READ Q : NEXT
220 D=Q : RESTORE
230 FOR I=1 TO 12 : READ Q : NEXT
310 FOR I=1 TO M : READ Q : NEXT
320 T=Q : RESTORE
340 F=Y-1600 : G=F-1
350 L=INT(G/4)-INT(G/100)+INT(G/400)+1
360 K=(L-INT(L/7)*7)+(F-INT(F/7)*7)+6+T
381 A=INT(Y/4)-INT(Y/100)+INT(Y/400)
383 B=INT((Y-1)/4)-INT((Y-1)/100)+INT((Y-1)/400)
384 N=A-B
385 IF M=2 THEN D=D+N
390 IF M>2 THEN K=K+N
400 W=(K-INT(K/7)*7)
410 DATA 31,28,31,30,31,30,31,31,30,31,30,31
430 DATA 0,3,3,6,1,4,6,2,5,0,3,5
500 :
510 OPEN4,4 : PRINT#4 : PRINT#4
520 PRINT#4,"       ";Y;" ";M
530 PRINT#4 : PRINT#4
540 PRINT#4,"SUN MON TEU WED THU FRI SAT"
550 PRINT#4 : PRINT#4
560 IF W=0 THEN 580
570 FOR I=1 TO W : PRINT#4,SPC(4);: NEXT
580 S=W
590 FOR I=1 TO D
595 S=S+1
600 A$=RIGHT$(" "+STR$(I)+"  ",4)
610 PRINT#4,A$;
620 IF S=7 THEN GOSUB 700
630 NEXT
640 PRINT#4 : PRINT#4 : PRINT#4
680 CLOSE4
690 END
700 :
710 PRINT#4 : PRINT#4
720 S=0
730 RETURN
```

C-64

パソコンデス・マッチ

理美ちゃんがコモドール-64にチャレンジ!!

アクエリアス・カレンダーを作ってみよう

```
       1983   10

SUN MON TEU WED THU FRI SAT

                          1

  2   3   4   5   6   7   8

  9  10  11  12  13  14  15

 16  17  18  19  20  21  22

 23  24  25  26  27  28  29

 30  31

       1983   11

SUN MON TEU WED THU FRI SAT

              1   2   3   4   5

  6   7   8   9  10  11  12

 13  14  15  16  17  18  19

 20  21  22  23  24  25  26

 27  28  29  30
```

```
       1983   12

SUN MON TEU WED THU FRI SAT

                  1   2   3

  4   5   6   7   8   9  10

 11  12  13  14  15  16  17

 18  19  20  21  22  23  24

 25  26  27  28  29  30  31
```

commodore CBM computer
commodore CBM dual drive floppy disk

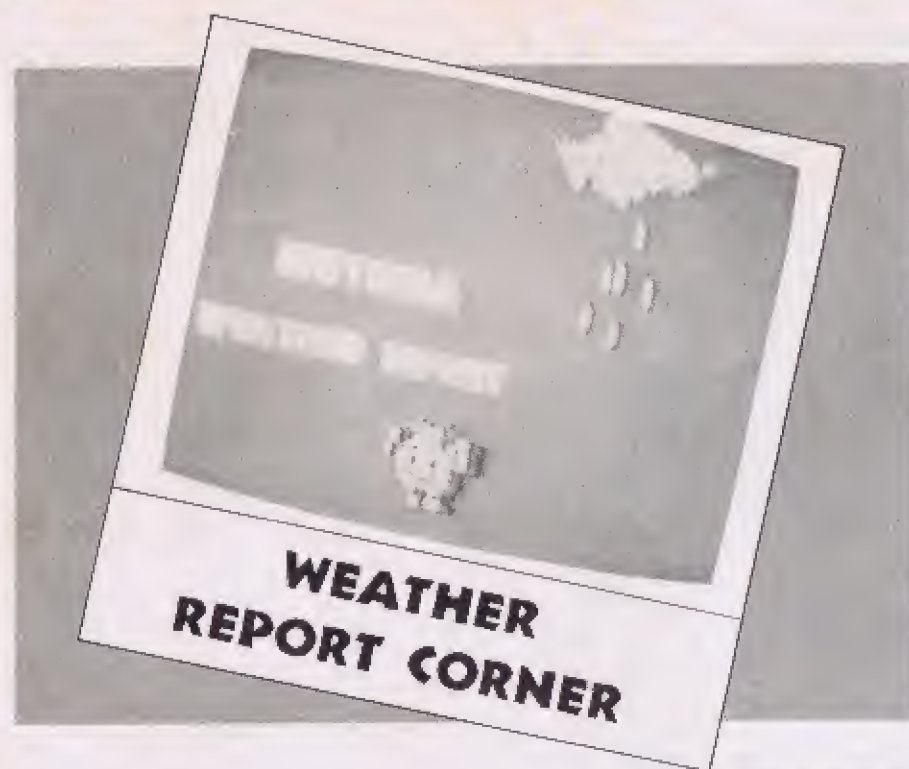
LIST UP
OK! コモドール64のオモシロプログラム・オン・パレード

ハチャメチャ
ゆかいに飛びはねる
大勢のトマトマト…


```
100 REM************************************
110 REM*                                  *
120 REM*          JUMPING TOMATO          *
130 REM*      WITH MACHINE LANGUAGE       *
140 REM*                                  *
150 REM* COPYRIGHT BY COMMODORE JAPAN *
160 REM*                                  *
170 REM*      DATE : 1982/11/20           *
180 REM************************************
190 VIC=13*4096
200 SID=13*4096+4*256
210 REM -- SCREEN SET
220 PRINT"■■";
230 POKEVIC+32,1:REM BORDER COLOR
240 POKEVIC+33,1:REM BACK G.COLOR
241 POKEVIC+35,2:POKEVIC+36,0:POKEVIC+34,5:POKEVIC+37,5:POKEVIC+38,0
242 POKEVIC+28,255
250 PRINT"■┌";:FORI=1TO37:PRINT"─";:NEXTI:PRINT"┐"
260 FORI=1TO23:PRINT"│";TAB(38);"│":NEXTI
270 PRINT"└";:FORI=1TO37:PRINT"─";:NEXTI:PRINT"┘";
280 PRINT"■■■■■■■■■";
290 PRINTTAB(1);:FORJ=1TO37:PRINT"■ ";:NEXTJ:PRINT
291 PRINT"■■                  MUSIC TOMATO              "
292 PRINT"■■                                            "
300 PRINT"■■■■■■■■";:REM COLOR IS RED
310 PRINTTAB(1);:FORJ=1TO37:PRINT"■ ";:NEXTJ:PRINT
311 PRINT"■■  1983 COPYRIGHT BY COMMODORE JAPAN  "
312 PRINT"■■                                            "
320 REM -- SET COORDINATE (X,Y)
330 FORI=0TO7:N=(7+I*4)*8
340 POKEVIC+I*2,N-INT(N/256)*256
350 READA:POKEVIC+I*2+1,A:NEXT I
360 REM -- SET SPRITE DATA
370 FORI=0TO62:READA
380 POKE8*256+I,A:NEXT I
390 REM -- SET SPRITE POINTERS
400 FORI=0TO7:POKE2040+I,32:NEXT I
410 REM -- SET COLOR
420 FORI=0TO7:READA
430 POKEVIC+39+I,A:NEXT I
440 REM -- SET SPRITE
```

```
450 POKEVIC+16,128:REM MSB OD X COORD.
460 POKEVIC+23,0:REM Y EXPAND OFF
470 POKEVIC+29,255:REM X EXPAND OFF
480 POKEVIC+27,0:REM BACKG. PRIORITY
490 POKEVIC+28,255:REM MULTI COLOR
500 REM -- SET SOUND
510 POKESID+24,15
520 POKESID+12,1*16+0
530 POKESID+13,1*16+4
540 REM -- MACHINE CODE SET
550 :
560 I=4*4096+4:J=0
570 READ A$:IF A$="-1" THEN640
580 N=ASC(MID$(A$,1,1)):GOSUB620:A=N*16
590 N=ASC(MID$(A$,2,1)):GOSUB620:A=A+N
600 POKE I+J,A
610 J=J+1:GOTO570
620 S=48:IF65<=NTHEN S=55
630 N=N-S:RETURN
640 REM -- MACHINE CODE ROUTINE CALL
650 :
680 SYS4*4096+4:REM ** CALL
690 REM -- Y COORD. & BACKG. PRIORITY
700 DATA 66,130, 55, 82,144, 90,136, 55
710 REM -- SPRITE DATA
720 DATA  0,  0,  0,  0,  0,  0,  0, 64, 64
730 DATA  0, 17,  0,  1, 85,  0,160, 20, 10
740 DATA160,105, 10,130,170,130,130,235,130
750 DATA138,170,162,170,235,170, 10,235,160
760 DATA 10,170,160, 10,235,160, 10,190,160
770 DATA  2,170,128,  2,130,128,  0,130,  0
780 DATA  0,162,128,  0,162,128,  0,  0,  0
790 REM -- SPRITE CLOR
800 DATA 8, 8, 8,8,8,8,8,8
810 :
820 REM MACHINE LANGUAGE CODE
830 :
840 REM *** SYMBOL ADDRESS TABLE
850 REM   YADR  = $4E
860 REM   BASE  = $50
870 REM   COUNT = $4000
880 REM   OFFSET= $4001
890 REM   YC1   = $4002
900 REM   YC2   = $4003
910 REM   VIC   = $D015
920 REM   SID1  = $D407
930 REM   SID2  = $D408
940 REM   SID3  = $D40B
950 REM   BGND  = $D01B
960 :
970 REM *** THIS ROUTINE START IN $4004
980 REM  CODE      :    LOC.   MNEMONIC  ; COMMENT
990 :
1000 DATA A9,FF    :REM          LDA #255
1010 DATA 8D,15,D0:REM           STA VIC ;ALL SPRITE ENABLE
1020 DATA A9,01    :REM          LDA #1
1030 DATA 85,4E    :REM          STA YADR
```

```
1040 DATA A9,D0     :REM        LDA #$D0 ;YADR=$D001
1050 DATA 85,4F     :REM        STA YADR+1
1060 DATA A9,00     :REM CNTSET LDA #0
1070 DATA 8D,00,40:REM          STA COUNT
1080 DATA AD,00,40:REM ADRSET   LDA COUNT
1090 DATA 0A        :REM        ASL A ; A=A*2
1100 DATA A8        :REM        TAY
1110 DATA 8D,01,40:REM          STA OFFSET
1120 DATA 0A        :REM        ASL A
1130 DATA 18        :REM        CLC
1140 DATA 69,F8     :REM        ADC #<INFO
1150 DATA 85,50     :REM        STA BASE
1160 DATA A9,00     :REM        LDA #0
1170 DATA 69,40     :REM        ADC #>INFO
1180 DATA 85,51     :REM        STA BASE+1 ;BASE=ADDRESS OF SYMBOL (INFO)
1190 DATA B1,4E     :REM        LDA (YADR),Y
1200 DATA 8D,02,40:REM          STA YC1 ;CURRENT Y COORDINATE
1210 DATA A0,03     :REM        LDY #3
1220 DATA 71,50     :REM        ADC (BASE),Y
1230 DATA 8D,03,40:REM          STA YC2 ;UPDATE  Y COORDINATE
1240 DATA A0,00     :REM        LDY #0
1250 DATA B1,50     :REM        LDA (BASE),Y
1260 DATA CD,03,40:REM          CMP YC2
1270 DATA 90,05     :REM        BCC RANGE
1280 DATA 8D,03,40:REM          STA YC2
1290 DATA B0,29     :REM        BCS SIGN ;JMP SIGN
1300 DATA A9,E1     :REM RANGE  LDA #255
1310 DATA CD,03,40:REM          CMP YC2
1320 DATA B0,2E     :REM        BCS BKGND1
1330 DATA 8D,03,40:REM          STA YC2
1340 DATA A9,11     :REM        LDA #$11
1350 DATA 8D,0B,D4:REM          STA SID3
1360 DATA AE,01,40:REM          LDX OFFSET
1370 DATA BD,18,41:REM          LDA SOUND,X
1380 DATA 8D,07,D4:REM          STA SID1
1390 DATA E8        :REM        INX
1400 DATA BD,18,41:REM          LDA SOUND,X
1410 DATA 8D,08,D4:REM          STA SID2 ; SOUND ON
1420 DATA A0,04     :REM        LDY #4
1430 DATA 20,E7,40:REM          JSR LOOP
1440 DATA A9,10     :REM        LDA #$10
1450 DATA 8D,0B,D4:REM          STA SID3 ; SOUND OFF
1460 DATA A9,00     :REM SIGN   LDA #0
1470 DATA A0,03     :REM        LDA #3
1480 DATA 38        :REM        SEC
1490 DATA F1,50     :REM        SBC (BASE),Y
1500 DATA 91,50     :REM        STA (BASE),Y ;BOUND SPEED RVS.
1510 DATA 4C,C7,40:REM          JMP UPDATE
1520 DATA AD,03,40:REM BKGND1   LDA YC2
1530 DATA C9,4D     :REM        CMP #77
1540 DATA 90,19     :REM        BCC BKGND2
1550 DATA C9,81     :REM        CMP #129
1560 DATA B0,15     :REM        BCS BKGND2
1570 DATA A0,01     :REM        LDY #1
1580 DATA B1,50     :REM        LDA (BASE),Y
1590 DATA F0,0F     :REM        BEQ BKGND2
1600 DATA AE,00,40:REM          LDX COUNT
```

```
1610 DATA BD,F0,40:REM        LDA BINRY,X
1620 DATA 0D,1B,D0:REM        ORA BGND
1630 DATA 8D,1B,D0:REM        STA BGND ;BACK GROUND PRIORITY ON
1640 DATA 4C,C7,40:REM        JMP UPDATE
1650 DATA AD,03,40:REM BKGND2 LDA YC2
1660 DATA C9,9D   :REM        CMP #157
1670 DATA 90,19   :REM        BCC BKGND3
1680 DATA C9,D1   :REM        CMP #209
1690 DATA B0,15   :REM        BCS BKGND3
1700 DATA A0,02   :REM        LDY #2
1710 DATA B1,50   :REM        LDA (BASE),Y
1720 DATA F0,0F   :REM        BEQ BKGND3
1730 DATA AE,00,40:REM        LDX COUNT
1740 DATA BD,F0,40:REM        LDA BINRY,X
1750 DATA 0D,1B,D0:REM        ORA BGND
1760 DATA 8D,1B,D0:REM        STA BGND ;BACK GROUND PRIORITY ON
1770 DATA 4C,C7,40:REM        JMP UPDATE
1780 DATA AE,00,40:REM BKGND3 LDX COUNT
1790 DATA BD,F0,40:REM        LDA BINRY,X
1800 DATA 49,FF   :REM        EOR #$FF
1810 DATA 2D,1B,D0:REM        AND BGND
1820 DATA 8D,1B,D0:REM        STA BGND ;BACK GROUND PRIORITY OFF
1830 DATA A0,03   :REM UPDATE LDY #3
1840 DATA 20,E7,40:REM        JSR LOOP
1850 DATA AD,03,40:REM        LDA YC2
1860 DATA AC,01,40:REM        LDY,OFFSET
1870 DATA 91,4E   :REM        STA (YADR),Y ;Y COORDINATE UPDATE
1880 DATA 18      :REM        CLC
1890 DATA AD,00,40:REM        LDA COUNT
1900 DATA 69,01   :REM        ADC #1
1910 DATA 8D,00,40:REM        STA COUNT
1920 DATA C9,08   :REM        CMP #8
1930 DATA 90,03   :REM        BCC JUMP
1940 DATA 4C,11,40:REM        JMP CNTSET
1950 DATA 4C,16,40:REM JUMP   JMP ADRSET
1960 DATA A2,FF   :REM LOOP   LDX #$FF ;WAITING SUB-ROUTINE
1970 DATA CA      :REM LOOP0  DEX
1980 DATA D0,FD   :REM        BNE LOOP0
1990 DATA 88      :REM        DEY
2000 DATA D0,F8   :REM        BNE LOOP
2010 DATA 60      :REM        RTS
2020
2030 DATA 01,02,04,08,10,20,40,80:REM BINRY  .BYTE 1,2,4,8,16,32,64,128
2040 DATA 42,01,00,04            :REM INFO   .BYTE 66,1,0,4
2050 DATA 82,00,01,03            :REM        .BYTE 130,0,1,3
2060 DATA 37,00,01,07            :REM        .BYTE 55,0,1,7
2070 DATA 52,01,00,04            :REM        .BYTE 82,1,0,4
2080 DATA 90,00,01,02            :REM        .BYTE 144,0,1,2
2090 DATA 5A,01,01,05            :REM        .BYTE 90,1,1,5
2100 DATA 88,01,00,03            :REM        .BYTE 136,1,0,3
2110 DATA 3C,00,01,04            :REM        .BYTE 60,0,1,4
2120 DATA 25,11,3F,13,9A,15,E3,16:REM SOUND  .BYTE 37,17,63,19,154,21,227,22
2130 DATA B1,19,D6,1C,5E,20,AF,22:REM        .BYTE 177,25,214,28,94,32,175,34
2140 DATA -1:REM  END MARK
```

雨がふります、雨がふる ピコピコッ、 トマトが傘をさす。


```
10 REM*************************************
20 REM*                                   *
30 REM*  TVK MUSIC TOMATO                 *
40 REM*       WETHER REPORT CORNER!!!     *
50 REM*                                   *
60 REM*  COPYRIGHT BY WATARU KANNO        *
70 REM*                  [19830404]       *
80 REM*                                   *
90 REM*************************************
100 REM START SETTING ROUTINE
110 PRINT"▗"
120 CLR
130 V=13*4096:C=2*4096:VR=C+8*256
140 POKE V+32,  3:REM BORDER COLOR
150 POKE V+33,  3:REM B.G. COLOR
200 REM SPRITE START SETING ROUTINE
210 POKE V+21,255:REMSPRITE ENABLE
220 POKE V+28,  4:REM MALTI SP#2 ON!
230 POKE V+39,  1:REM COLOR #0=WHITE
240 POKE V+40,  1:REM COLOR #1=WHITE
250 POKE V+41, 10:REM COLOR #2=L.RED
260 POKE V+42,  7:REM COLOR #3=YELL.
270 POKE V+43,  1:REM COLOR #4=WHITE
280 POKE V+44,  6:REM COLOR #5=BLUE.
290 POKE V+45,  1:REM COLOR #6=WHITE
310 POKE V+29,167:REM X-EXPAND
320 POKE V+23,247:REM Y-EXPAND
330 POKE 2040, 34:REM DATA SPACE SP#0
340 POKE 2041, 35:REM DATA SPACE SP#1
350 POKE 2042, 32:REM DATA SPACE SP#2
360 POKE 2043, 33:REM DATA SPACE SP#3
370 POKE 2044, 36:REM DATA SPACE SP#4
380 POKE 2045, 37:REM DATA SPACE SP#5
390 POKE 2046, 36:REM DATA SPACE SP#6
410 FOR I=32 TO 37
420 FOR J= 0 TO (63-1)
430 READ A:POKE I*64+J, A
440 NEXT J,I
450 POKE V+37,  5:REM SP MALTI#0
460 POKE V+38,  0:REM SP MALTI#1
999 REM DATA STATEMENT ROUTINE
```

```
1000 REM SP#2 DATA "TOMATO"
1010 DATA  0,  0,  0,  0,  0,  0
1020 DATA  0, 64, 64,  0, 17,  0
1030 DATA  1, 85,  0,160, 20, 10
1040 DATA160,105, 10, 30,170, 30
1050 DATA130,235,138,138,170,162
1060 DATA 70,235,170, 10,235,160
1070 DATA 10,170,160, 10,235,160
1080 DATA 10,190,160,  2,170,128
1090 DATA  2,130,128,  0,130,  0
1100 DATA  0,162,128,  0,162,128
1110 DATA  0,  0,  0
1200 REM SP#3 DATA "THE SUN"
1210 DATA  0,  0,  0,  0,  0,  0
1220 DATA  1,255,224,  7,255,224
1230 DATA 31,255,248, 63,255,252
1240 DATA 63,255,254,127,255,254
1250 DATA127,255,254,127,255,254
1260 DATA127,255,254,127,255,254
1270 DATA127,255,254,127,255,254
1280 DATA127,255,254, 63,255,252
1290 DATA 63,255,248, 31,255,248
1300 DATA  7,255,224,  1,255,128
1310 DATA  0,  0,  0
1400 REM SP#0 DATA "CLOUD LEFT"
1410 DATA  0,  0,  0,  0,  0,  0
1420 DATA  0,  0,  0,  0,  0, 24
1430 DATA  0,  0,253,  0,  0,253
1440 DATA  0, 31,255,  0, 47,255
1450 DATA 90,183,255, 39,219,255
1460 DATA 63,237,255, 31,191,255
1470 DATA  7,223,255,  7,239,255
1480 DATA  3,247,255,  1,227,255
1490 DATA  1,227,255,  0,192,127
1500 DATA  0,  0, 15,  0,  0,  0
1510 DATA  0,  0,  0
1600 REM SP#1 DATA "CLOUD RIGHT"
1610 DATA  0,  0,  0,  0,  0,  0
1620 DATA 28,  0,  0,127,  0,  0
1630 DATA255,192,  0,127,236,  0
1640 DATA191,222,  0,255,191,128
1650 DATA255,127,128,255,255,192
1660 DATA255,255,192,253,255,128
1670 DATA254,255,  0,255,157,  0
1680 DATA254,  0,  0,240,  0,  0
1690 DATA224,  0,  0,  0,  0,  0
1700 DATA  0,  0,  0,  0,  0,  0
1710 DATA  0,  0,  0
1800 REM SP#4 & #6 DATA "RAIN"
1810 DATA  0,  0,  0,  0, 16,  0
1820 DATA  0, 16,  0,  0, 56,  0
1830 DATA  0,124,  0,  0,124,  0
1840 DATA  0,124,  0,  0,124,  0
1850 DATA  0, 56,  0,  0,  0,  0
1860 DATA 16,  0,  0, 16,  0,  0
1870 DATA 56,  0, 16,124,  0, 16
1880 DATA124,  0, 56,124,  0,124
```

```
1890 DATA124,  0,124, 56,  0,124
1900 DATA  0,  0,124,  0,  0, 56
1910 DATA  0,  0,  0
2000 REM SP#5 DATA "UMBRELLA"
2010 DATA  0,  0,  0,  0,  0,  0
2020 DATA  0,  8,  0,  1,255,128
2030 DATA  7,255,244, 31,255,248
2040 DATA 31,255,248, 63,255,252
2050 DATA127,255,254,127,255,254
2060 DATA103, 60,230, 66, 24, 66
2070 DATA  0,  8,  0,  0,  8,  0
2080 DATA  0,  8,  0,  0,  8,  0
2090 DATA  0,  8,  0,  0,  8,  0
2100 DATA  0,  8,  0,  0,  8,  0
2110 DATA  0,  0,  0
2999 REM SPRITE MOVEMENT ROUTINE
3000 POKE V+6,230:POKE V+7,60
3010 POKE V+1,65:POKE V+3,65
3020 FORI=0TO80
3030 POKE V+0,I:POKEV+2,I+48
3040 NEXT
3050 POKE V+5,200
3060 FORI=1TO120
3070 POKEV+4,255-I:POKEV+0,I+80:POKEV+2,I+128
3080 NEXT I
3089 POKE V+8,245:POKE V+12,230
3090 POKE V+32,15:POKEV+33,15:FORI=66TO235:POKE V+9,I:POKE V+13,I+20:NEXT
3100 GOTO3140:FORI=129TO345
3110 IF I>=256THEN POKE V+16,4:POKE V+4,I-255:POKE V+9,66:GOSUB5000:GOTO3130
3120 POKE V+4,I:POKE V+9,I:POKE V+13,I-35
3130 NEXTI
3140 POKE V+16,0
3150 FORI=0TO245
3151 IF I<=100THEN POKE V+4,I:POKE V+13,0:POKE V+9,0:GOTO3170
3160 POKE V+4,I:POKEV+9,I-30:POKEV+13,I-100
3170 NEXT I
3180 POKE V+10,225:POKE V+11,180
3200 PRINT"[illegible]MU-TOMA"
3210 PRINT"[illegible]WETHER REPORT"
3230 GETA$:IF A$=""THEN3230:STOP
5000 IF I>=290 THENPOKE V+13,66:RETURN
```

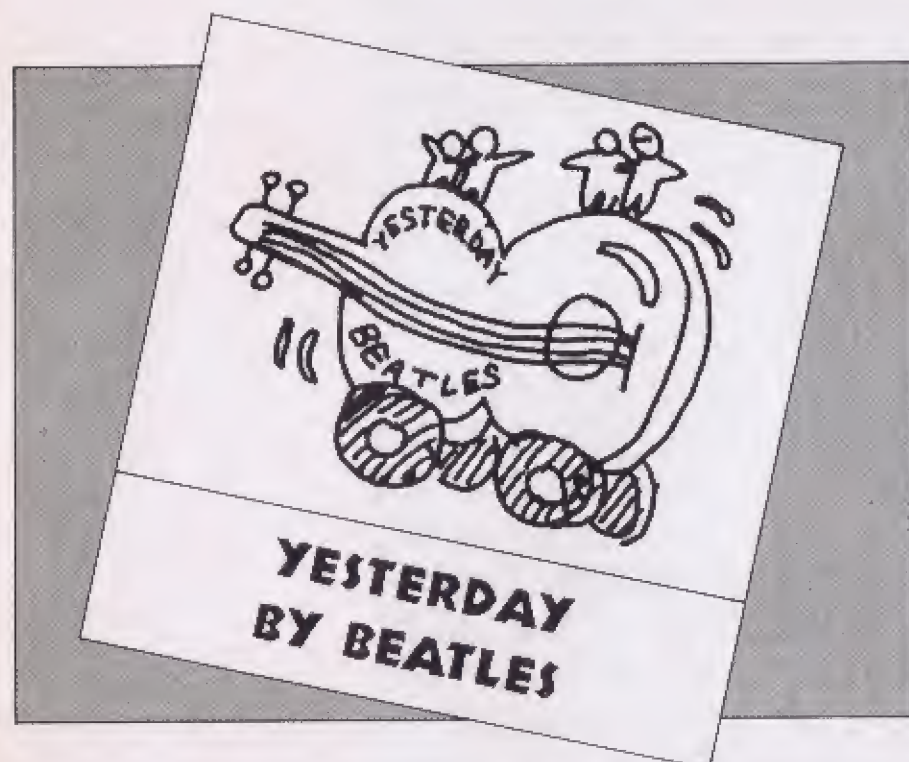
クルックルッ
地球のまわりをかくれんぼ
して、バーイバイ。


```
10 REM*********************************
20 REM*                               *
30 REM* TVK MUSIC TOMATO ENDING       *
40 REM*             "EARTH & TOMATO"  *
50 REM*                               *
60 REM*  COPYRIGHT BY COMMODORE JAPAN *
70 REM*                [19830404]     *
80 REM*                               *
90 REM*********************************
100 DIM X(63),Y(63),XX(63),YY(63),B(63)
101 A=COS(160*π/180):B=SIN(160*π/180)
102 FOR I=0TO63
104 X(I)=100*SIN(2*π*I/63)
106 Y(I)=10*COS(2*π*I/63)
108 XX(I)=INT(X(I)*A-Y(I)*B+160):YY(I)=INT(X(I)*B+Y(I)*A+128)
110 NEXT
120 B(0)=1:B(1)=1:B(2)=1:B(3)=1:B(4)=0:B(5)=0:B(6)=0:B(7)=0:B(8)=0:B(9)=0
122 B(18)=0:B(19)=0:B(20)=0:B(21)=0:B(22)=0:B(23)=0:B(24)=0:B(25)=0:
123 B(26)=0:B(27)=0:B(28)=0:B(29)=0:B(30)=0:B(31)=0
124 B(32)=0:B(33)=0:B(34)=0:B(35)=0:B(36)=0:B(37)=0:B(38)=0:B(39)=0:B(40)=0
125 B(41)=0:B(42)=0:B(43)=0:B(44)=0:B(45)=0:B(46)=0:B(47)=0:B(48)=0:B(49)=0
126 B(50)=0:B(51)=0:B(52)=0:B(53)=0:B(54)=0:B(55)=0:B(56)=0:B(57)=0:B(58)=0
127 B(58)=1:B(59)=1:B(60)=1:B(61)=1:B(62)=1:B(63)=1
199 :
200 REM * INITIALIZE *****************
210 :
220 VIC=13*4096:MSB=VIC+16
230 CB=3*4096+8*256:SB=3*4096+14*256
240 VRAM=4*256:CRAM=13*4096+8*256
250 M0%=254:M1%=253:M2%=251:M3%=247:M4%=239:M5%=223:M6%=191:M7%=127
260 S0%=1:S1%=2:S2%=4:S3%=8:S4%=16:S5%=32:S6%=64:S7%=128
270 :
300 REM * SCREEN PAINTING *************
301 :
310 PRINT"WRITING DATA"
320 POKE VIC+24,21 :REM CB=$1000
330 POKE VIC+17,16 :REM SCREEN SET
340 :
350 FOR I=0TO120*8-1:READA:POKECB+I,A:PRINT""I:NEXT:REM CHARACTER WRITING
360 FOR I=0TO1*63-1:READA:POKESB+I,A:PRINT""I:NEXT:REM SPRITE WRITING
370 POKE VIC+33,12           :REM SCREEN COLOR
```

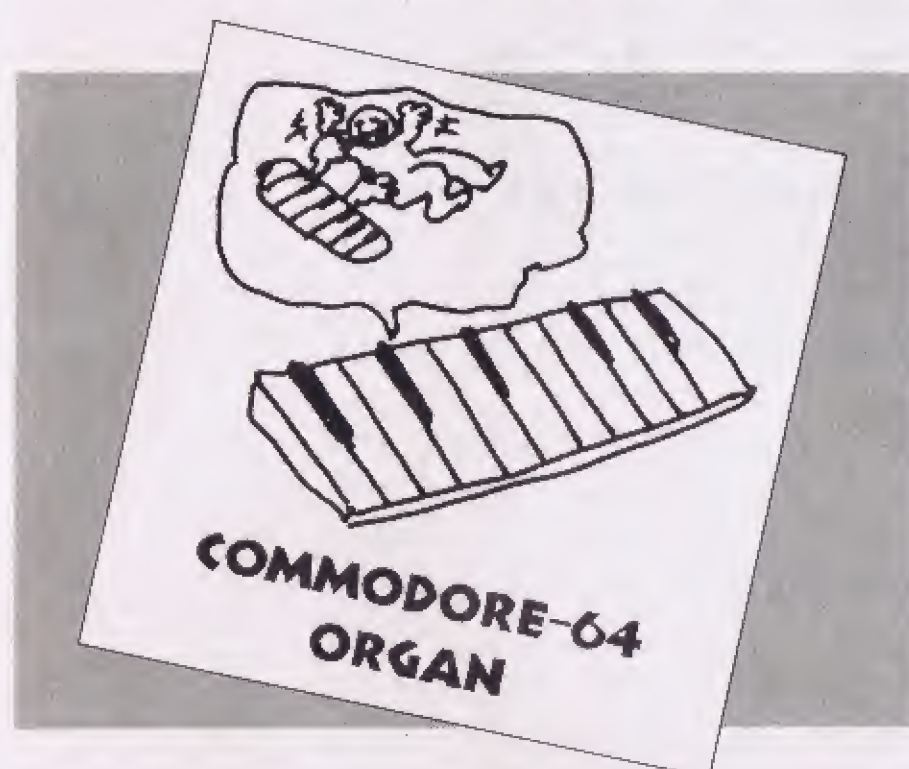
```
372 POKE VIC+32,12          :REM BORDER COLOR
380 FOR I=0TO999:POKECR+I,8+1:NEXT:REM 11COLOR
385 POKE VIC+35,6           :REM BACKGROUND COLOR#2 :BLUE
390 :
400 FORI=0  TO293:POKEVR+I,0     :NEXT
402 FORI=294TO305:POKEVR+I,I-294:NEXT
404 FORI=306TO333:POKEVR+I,0     :NEXT
406 FORI=334TO345:POKEVR+I,I-322:NEXT
408 FORI=346TO373:POKEVR+I,0     :NEXT
410 FORI=374TO385:POKEVR+I,I-350:NEXT
412 FORI=386TO413:POKEVR+I,0     :NEXT
414 FORI=414TO425:POKEVR+I,I-378:NEXT
416 FORI=426TO453:POKEVR+I,0     :NEXT
418 FORI=454TO465:POKEVR+I,I-406:NEXT
420 FORI=466TO493:POKEVR+I,0     :NEXT
422 FORI=494TO505:POKEVR+I,I-434:NEXT
424 FORI=506TO533:POKEVR+I,0     :NEXT
426 FORI=534TO545:POKEVR+I,I-462:NEXT
428 FORI=546TO573:POKEVR+I,0     :NEXT
430 FORI=574TO585:POKEVR+I,I-490:NEXT
432 FORI=586TO613:POKEVR+I,0     :NEXT
434 FORI=614TO625:POKEVR+I,I-518:NEXT
436 FORI=626TO653:POKEVR+I,0     :NEXT
438 FORI=654TO665:POKEVR+I,I-546:NEXT
440 FORI=666TO999:POKEVR+I,0     :NEXT
450 POKE VIC+24,31
460 POKE VIC+22,16          :REM MULTI COLOR MODE
470 :
500 REM * SPRITE MOVING **************
501 :
512 :
520 POKE VIC+27,1  :REM SP VS BACK PRIORITY
525 POKE VIC+23,255:REM SP Y-EXP %00000001
530 POKE VIC+29,255:REM SP X-EXP %00000001
540 POKE VIC+28,255:REM SP MULTI %00000001
550 POKE VIC+37,13 :REM SP MULTI COLOR#1 01COLOR
555 POKE VIC+38,0' :REM SP MULTI COLOR#2 11COLOR
560 POKE VIC+39,10 :REM SP0 COLOR
578 :
590 POKE 2040,248
600 POKE VIC+21,1  :REM SP ENABLE
610 FORI=0TO63:POKEVIC,XX(I):POKEVIC+1,YY(I):POKEVIC+27,B(I)
615 NEXT
620 POKE2040,248:GOTO610
1998 :
1999 REM CHARACTER FONT **************
2000 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA0
2001 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA1
2002 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA2
2003 DATA  0,  0,  0,  0,  0,  0,  3, 63:REM CHA3
2004 DATA  0,  0,  0,  0, 15, 63,255,254:REM CHA4
2005 DATA  0,  0,  0,  0,255,239,171,170:REM CHA5
2006 DATA  0,  0,  0,  0,255,255,255,255:REM CHA6
2007 DATA  0,  0,  0,  0,240,252,255,255:REM CHA7
2008 DATA  0,  0,  0,  0,  0,  0,192,252:REM CHA8
2009 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA9
2010 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA10
```

```
2011 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA11
2012 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA12
2013 DATA  0,  0,  0,  0,  0,  3,  3, 15:REM CHA13
2014 DATA  0,  3, 15, 63,255,255,255,255:REM CHA14
2015 DATA255,255,255,255,255,255,255,251:REM CHA15
2016 DATA255,254,250,190,170,254,255,255:REM CHA16
2017 DATA171,175,175,175,191,191,191,255:REM CHA67
2018 DATA255,255,255,191,171,171,175,171:REM CHA18
2019 DATA251,254,254,255,255,255,255,255:REM CHA19
2020 DATA255,255,190,170,170,250,254,255:REM CHA20
2021 DATA  0,128,160,168,170,170,170,170:REM CHA21
2022 DATA  0,  0,  0,  0,  0,128,128,160:REM CHA22
2023 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA23
2024 DATA  0,  0,  0,  0,  0,  3,  3,  2:REM CHA24
2025 DATA 15, 63, 63,255,255,255,255,254:REM CHA25
2026 DATA255,255,255,255,250,234,170,171:REM CHA26
2027 DATA250,254,250,234,234,170,170,170:REM CHA26
2028 DATA187,170,170,170,170,186,254,254:REM CHA26
2029 DATA250,255,191,171,170,170,170,186:REM CHA26
2030 DATA170,250,234,170,171,170,170,170:REM CHA26
2031 DATA191,255,175,255,255,254,186,170:REM CHA26
2032 DATA255,255,251,251,187,170,170,171:REM CHA26
2033 DATA170,254,255,255,255,255,255,255:REM CHA26
2034 DATA160,168,232,250,254,254,255,255:REM CHA34
2035 DATA  0,  0,  0,  0,  0,128,128,160:REM CHA35
2036 DATA 10, 14, 15, 63, 63, 63, 63, 63:REM CHA36
2037 DATA191,171,170,250,191,170,250,255:REM CHA37
2038 DATA191,255,174,234,251,255,191,174:REM CHA38
2039 DATA234,170,170,234,250,235,171,170:REM CHA39
2040 DATA187,254,191,255,190,250,250,170:REM CHA40
2041 DATA250,234,250,170,170,170,171,175:REM CHA41
2042 DATA170,170,170,171,191,255,255,254:REM CHA42
2043 DATA170,170,190,254,250,234,170,170:REM CHA43
2044 DATA170,170,170,170,170,170,170,170:REM CHA44
2045 DATA255,190,186,170,170,170,170,170:REM CHA45
2046 DATA191,175,191,255,191,191,175,174:REM CHA46
2047 DATA240,240,240,252,252,252,252,252:REM CHA47
2048 DATA 63,255,255,190,174,170,170,174:REM CHA48
2049 DATA255,254,250,234,170,170,170,170:REM CHA49
2050 DATA170,170,170,170,170,170,170,170:REM CHA50
2051 DATA170,170,170,170,170,170,170,170:REM CHA51
2052 DATA170,170,170,171,171,175,171,171:REM CHA52
2053 DATA191,191,251,251,239,239,239,239:REM CHA53
2054 DATA254,250,250,234,234,170,170,170:REM CHA54
2055 DATA170,170,170,170,170,170,170,170:REM CHA55
2056 DATA170,170,170,171,175,191,255,255:REM CHA56
2057 DATA170,170,234,250,255,255,255,255:REM CHA57
2058 DATA170,170,170,170,170,234,250,254:REM CHA58
2059 DATA188,174,170,170,170,170,170,170:REM CHA59
2060 DATA191,191,250,254,255,255,254, 58:REM CHA60
2061 DATA175,191,175,171,191,175,191,191:REM CHA61
2062 DATA234,250,255,255,255,255,255,255:REM CHA62
2063 DATA170,170,170,234,238,238,239,239:REM CHA63
2064 DATA175,175,191,191,255,191,191,191:REM CHA64
2065 DATA239,255,255,255,255,255,255,255:REM CHA65
2066 DATA234,234,250,250,251,255,255,255:REM CHA66
2067 DATA171,175,175,175,191,191,191,255:REM CHA67
```

```
2068 DATA255,255,255,255,254,254,250,254:REM CHA68
2069 DATA255,250,234,170,170,170,170,170:REM CHA69
2070 DATA190,171,170,170,170,170,170,171:REM CHA70
2071 DATA170,170,170,170,170,170,170,171:REM CHA71
2072 DATA 62, 58, 42, 42, 42, 11, 15, 15:REM CHA72
2073 DATA175,191,255,187,170,187,255,255:REM CHA73
2074 DATA255,254,234,175,255,255,255,255:REM CHA74
2075 DATA190,175,175,171,235,235,234,250:REM CHA75
2076 DATA175,170,239,250,254,255,255,191:REM CHA76
2077 DATA255,255,170,171,255,255,255,254:REM CHA77
2078 DATA255,255,255,255,255,255,255,191:REM CHA78
2079 DATA255,255,251,234,234,235,171,175:REM CHA79
2080 DATA254,255,255,255,255,255,255,255:REM CHA80
2081 DATA170,234,234,250,250,234,235,170:REM CHA81
2082 DATA175,191,175,171,170,139,239,255:REM CHA82
2083 DATA238,248,252,252,252,240,240,240:REM CHA83
2084 DATA 15,  3,  3,  0,  0,  0,  0,  0:REM CHA84
2085 DATA255,255,255,255,255, 63, 63, 15:REM CHA85
2086 DATA239,171,238,170,238,254,250,254:REM CHA86
2087 DATA250,170,170,170,170,170,170,170:REM CHA87
2088 DATA175,171,170,170,170,170,170,171:REM CHA88
2089 DATA255,250,234,170,170,170,170,171:REM CHA89
2090 DATA190,186,170,171,175,191,255,254:REM CHA90
2091 DATA191,191,255,255,255,250,234,170:REM CHA91
2092 DATA255,254,250,234,170,175,171,186:REM CHA92
2093 DATA170,171,170,171,239,255,255,255:REM CHA93
2094 DATA191,255,255,255,255,252,252,240:REM CHA94
2095 DATA240,192,192,  0,  0,  0,  0,  0:REM CHA95
2096 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA96
2097 DATA 15,  2,  2,  0,  0,  0,  0,  0:REM CHA97
2098 DATA250,234,170,170, 42, 10,  2,  0:REM CHA98
2099 DATA170,170,170,170,170,170,170,170:REM CHA99
2100 DATA175,171,170,170,170,170,170,170:REM CHA100
2101 DATA239,255,254,170,170,170,170,170:REM CHA101
2102 DATA254,234,171,170,170,174,174,191:REM CHA102
2103 DATA186,254,255,190,255,191,255,255:REM CHA103
2104 DATA254,191,190,175,190,255,255,255:REM CHA104
2105 DATA191,191,191,191,172,240,192,  0:REM CHA104
2106 DATA240,192,192,  0,  0,  0,  0,  0:REM CHA106
2107 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA107
2108 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA108
2109 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA109
2110 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA110
2111 DATA 42,  2,  0,  0,  0,  0,  0,  0:REM CHA111
2112 DATA170,170, 42, 10,  0,  0,  0,  0:REM CHA112
2113 DATA170,171,175,191,  0,  0,  0,  0:REM CHA113
2114 DATA255,255,255,255,  0,  0,  0,  0:REM CHA114
2115 DATA255,255,252,240,  0,  0,  0,  0:REM CHA115
2116 DATA252,192,  0,  0,  0,  0,  0,  0:REM CHA116
2117 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA117
2118 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA118
2119 DATA  0,  0,  0,  0,  0,  0,  0,  0:REM CHA119
2998 :
2999 REM * SPRITE PATTERN DATA ********
3000 REM * SPRITE #0 TOMATO ***********
3001 DATA  0,  0,  0,  0,  0,  0,  0, 64, 64
3002 DATA  0, 17,  0,  1, 85,  0,160, 20, 10
```

```
3003 DATA160,105, 10, 30,170, 30,130,235,138
3004 DATA138,170,162,170,235,170, 10,235,160
3005 DATA 10,170,160, 10,235,160, 10,190,160
3006 DATA  2,170,128,  2,130,  0,  0,130,  0
3007 DATA  0,162,128,  0,162,128,  0,  0,  0
3008 REMDATA0
```

ヤッタネ！
僕だけの誰にもマネできない
“イエスタディ”

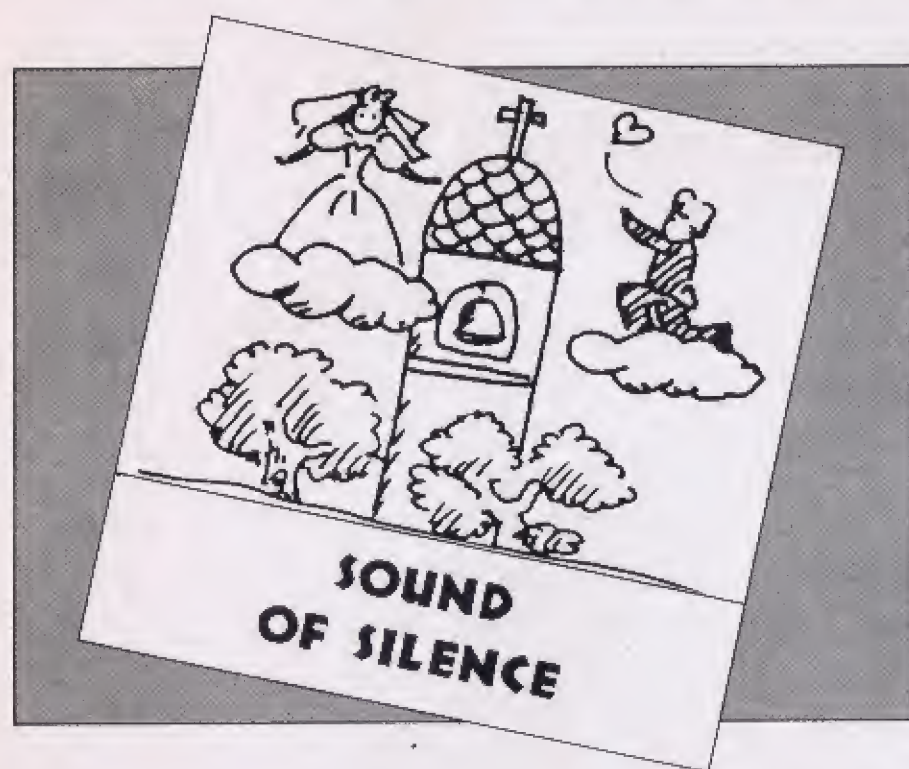

```
10 REM *****************,
20 REM *    YESTERDAY    *
30 REM * BY  THE BEATLES *
40 REM *FOR COMMODORE -64*
50 REM *   (C) '83.4.5   *
60 REM *  BY  H. TANAKA  *
70 REM ******************
80 :
90 SID=54272:I=79:DIMH1(I),H2(I),L1(I),L2(I),D(I),R(6),S(6)
100 FORI=0TO28:POKESID+I,0:NEXT
110 POKESID+24,15
120 POKESID+5,9:POKESID+6,0
130 POKESID+12,13:POKESID+13,0
140 FORI=1TO6:READR(I),S(I):NEXT
150 FORI=1TO78:READH1(I),L1(I),H2(I),L2(I),D(I):NEXT
160 REM PLAY
170 FORJ=1TO7
180 FORI=R(J)TOS(J)
190 IFL1(I)<>-1THENPOKESID,L1(I):POKESID+1,H1(I)
200 IFL2(I)<>-1THENPOKESID+7,L2(I):POKESID+8,H2(I)
210 IFL1(I)<>-1THENPOKESID+4,33
220 IFL2(I)<>-1THENPOKESID+11,17
230 IFL1(I)=-1THENPOKESID+16,0:POKESID+16,0:POKESID+16,0
240 IFL2(I)=-1THENPOKESID+16,0:POKESID+16,0
250 FORT=1TO300*(D(I)-.1):NEXT
260 IFL1(I+1)<>-1THENPOKESID+4,32
270 IFL2(I+1)<>-1THENPOKESID+11,16
280 NEXT:NEXT
290 FORI=0TO28:POKESID+I,0:NEXT:END
300 REM PARA-METER
310 DATA1,34,1,34,35,70,1,70,1,34,71,78
320 REM MUSIC DATA
330 REM [1] (1-34)
340 DATA28,69,12,152,.45,25,48,-1,-1,.45,25,48,-1,-1,2,-1,-1.12,152,1
```

```
350 DATA0,0,11,227,1,31,188,-1,-1,.45,35,158,-1,-1,.45,39,251,0,0,.45
360 DATA42,92,-1,-1,.45,47,140,-1,-1,.45,50,95,-1,-1,.45
370 DATA47,140,10,151,1,-1,-1,25,48,.45,42,92,-1,-1,.45,42,92,21,46,1
380 DATA-1,-1,18,222,1,0,0,16,207,1,42,92,-1,-1,1,37,189,18,222,.45
390 DATA33,159,-1,-1,.45,31,188,-1,-1,.45,28,69,-1,-1,.45
400 DATA33,159,12,152,1,-1,-1,18,222,.45,31,188,-1,-1,.45,31,188,12,152,1
410 DATA28,69,14,35,1,25,48,10,151,1,31,188,-1,-1,1,28,69,14,35,1,21,46,-1,-1,1
420 DATA25,48,16,207,1.45,31,188,-1,-1,.45,31,188,12,152,2
430 REM [2] (35-70)
440 DATA31,188,21,46,2,31,188,19,253,1,-1,-1,15,222,1,42,92,10,151,1
450 DATA47,140,18,222,1,50,95,16,207,1,47,140,18,222,.45,42,92,-1,-1,.45
460 DATA47,140,14,35,.45,-1,-1,28,69,.45,-1,-1,33,159,.45,42,92,42,92,.45
470 DATA37,189,18,222,1,42,92,-1,-1,1,31,188,12,152,2,0,0,31,188,.45
480 DATA-1.-1,25,48,.45,-1,-1,18,222,.45,-1,-1,16,207,.45
490 DATA31,188,15,222,2,31,188,19,253,2,42,92,21,46,1,47,140,18,222,1
500 DATA50,95,16,207,1,47,140,18,222,.45,42,92,-1,-1,.45,47,140,14,35,.45
510 DATA-1,-1,28,69,.45,-1,-1,33,159,.45,42,92,42,92,.45,37,189,18,222,1
520 DATA47,140,-1,-1,1,50,95,25,48,1,37,189,-1,-1,1
530 DATA33,159,-1,-1,1,31,188,-1,-1,1
540 REM [3] (71-78)
550 DATA25,48,10,151,1.2,31,188,-1,-1,1.2,28,69,14,35,1.2,21,46,-1,-1,1.2
560 DATA25,48,16,207,1.2,-1,-1,12,152,.55,31,188,-1,-1,.55,31,188,25,48,3
```

オルガンにはオルガンの意地があるんだ！チャレンジしちゃお。


```
100 REM ****************************
110 REM *   COMMODORE-64  ORGAN    *
120 REM *                          *
130 REM *         MUSIC            *
140 REM *           &              *
150 REM *  PROGRAMABLE CHARACTER   *
160 REM *                          *
170 REM *     DATE : 82/11/20      *
180 REM ****************************
190 VIC=13*4096:SID=13*4096+4*256
200 VR=4*256:CR=13*4096+8*256
210 GOSUB480:REM MESSAGE
220 GOSUB590:REM SOUND SET
230 GOSUB660:REM PRG.CHAR SET
240 POKESID+24,15:REM VOLUME SET
250 POKESID+12,0*16+12:REM A/D SET
260 POKESID+13,2*16+0 :REM S/R SET
270 :
```

```
280 REM **** SOUND CONTROL ****
290 G$="":GETG$:IFG$=""THEN290
300 P=ASC(G$)-65:REM KEY POSITION
310 IFP<0OR25<PTHEN290
320 IFSD(P,0)=0ANDSD(P,1)=0THEN290
330 POKESID+11,1*16+1:REM  GATE OPEN
340 POKESID+7,SD(P,0):REM  LOW SET
350 POKESID+8,SD(P,1):REM  HI  SET
360 Q=SD(P,2)+SD(P,3)*40:REM MARK PRINT POSITION
370 A=24:B=0:IF SD(P,3)=8 THEN A=25:B=1
380 POKE VR+Q,A:REM  MARK PRINT
390 POKE CR+Q,8+B:REM  MARK COLOR
400 FORI=1TO256:NEXTI:REM WAITING
410 POKESID+11,0:REM  GATE CLOSE
420 POKESID+7,0 :REM  OFF
430 POKESID+8,0 :REM  OFF
440 A=8:IF SD(P,3)=8 THEN A=9
450 POKE VR+Q,A:REM  MARK OFF
460 POKE CR+Q,8:REM  COLOR OFF
470 GOTO290
480 :
490 PRINT"[illegible]";CHR$(14);
500 PRINTTAB(10);"[illegible]ORGAN  KEYBOARD"
510 PRINT"[illegible]"
520 PRINT"[illegible] COMPUTER [illegible]"
530 PRINT"[illegible]KEY [illegible]"
540 PRINT"[illegible]PROGRAM [illegible] RUN/STOP KEY [illegible]"
550 PRINT"[illegible]RESTORE KEY [illegible]"
560 PRINTTAB(6);"[illegible]RETURN KEY [illegible]"
570 G$="":GETG$:IFG$<>CHR$(13)THEN570
580 RETURN
590 :
600 REM SOUND CONSTANT SET
610 DIM SD(25,3)
620 FORI=0TO25:SD(I,0)=0:SD(I,1)=0:NEXTI
630 FORI=1TO14:READ A:FORJ=0TO3
640 READ B:SD(A,J)=B:NEXTJ,I
650 RETURN
660 :
670 REM  PRG.CHAR (MULTI COLOR)
680 PRINT"[illegible]";TAB(10);"WRITING DATA :"
690 FORI=0TO25:P=8*256+8*I
700 FORJ=0TO7:PRINT"[illegible]";TAB(25);I*8+J
710 READ A:POKE P+J,A
720 NEXT J,I
730 :
740 POKEVIC+22,PEEK(VIC+22)OR16:REM MULTI-COLOR SET
750 POKEVIC+24,18:REM VIDEO=$0400,C-BASE=$0800
760 POKEVIC+32,6:REM BORDER COLOR
770 POKEVIC+33,6:REM BACK #0 COLOR
780 POKEVIC+34,1:REM BACK #1 COLOR
790 POKEVIC+35,0:REM BACK #2 COLOR
800 PRINT"[illegible]"
810 :REM  PIANO-KEY PRINT
820 FORI=0TO9:FORJ=0TO23:READ A
830 POKEVR+8+J+(I+6)*40,A
840 POKECR+8+J+(I+6)*40,8+4
```

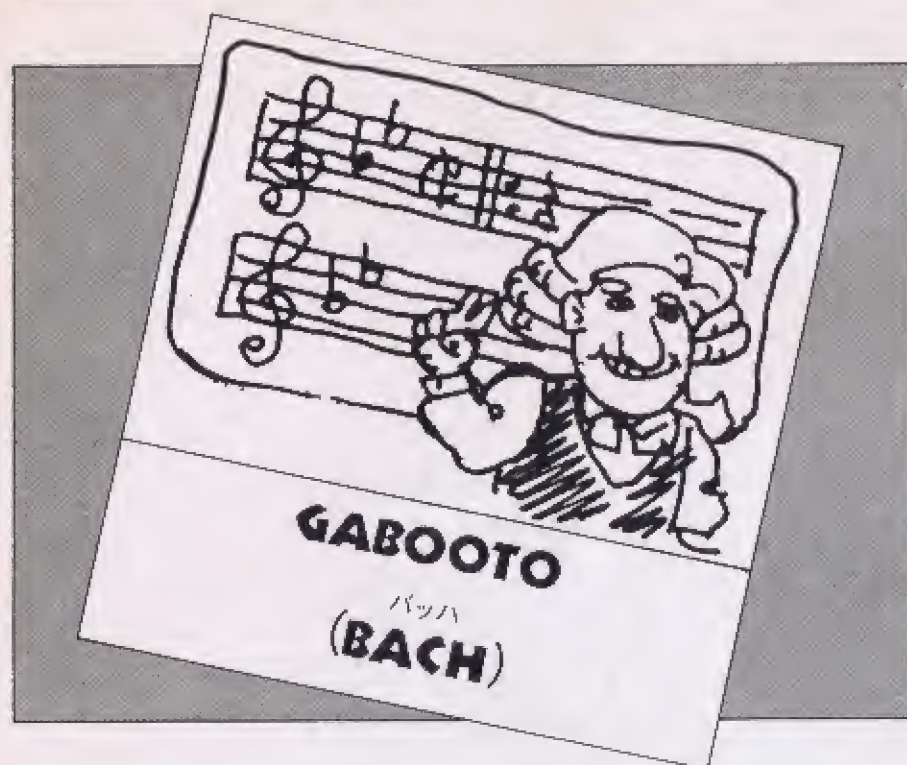
```
850 NEXT J,I
860 RETURN
870 :
880 REM  SOUND DATA
890 DATA   0, 37, 17,  9, 13:REM A KEY
900 DATA  22, 42, 18, 10,  8:REM W KEY
910 DATA  18, 63, 19, 12, 13:REM S KEY
920 DATA   4,100, 20, 13,  8:REM E KEY
930 DATA   3,154, 21, 15, 13:REM D KEY
940 DATA   5,227, 22, 18, 13:REM F KEY
950 DATA  19, 63, 24, 19,  8:REM T KEY
960 DATA   6,117, 25, 21, 13:REM G KEY
970 DATA  24, 56, 27, 22,  8:REM Y KEY
980 DATA   7,214, 28, 24, 13:REM H KEY
990 DATA  20,141, 30, 25,  8:REM U KEY
1000 DATA   9, 94, 32, 27, 13:REM J KEY
1010 DATA  10, 75, 34, 30, 13:REM K KEY
1020 DATA  14, 85, 36, 31,  8:REM O KEY
1030 :
1040 REM PIANO-KEY DATA (PRG.CHAR)
1050 DATA 149,149,149,149,149,149,149,149
1060 DATA  86, 86, 86, 86, 86, 86, 86, 86
1070 DATA 170,170, 85, 85, 85, 85, 85, 85
1080 DATA  85, 85, 85, 85, 85, 85,170,170
1090 DATA 149,149,149,149,149,149,170,170
1100 DATA  86, 86, 86, 86, 86, 86,170,170
1110 DATA 170,170,149,149,149,149,149,149
1120 DATA 170,170, 86, 86, 86, 86, 86, 86
1130 DATA  85, 85, 85, 85, 85, 85, 85, 85
1140 DATA 170,170,170,170,170,170,170,170
1150 :
1160 REM CHARACTER DATA (PRG.CHAR)
1170 DATA 105,105,150,150,150,170,150,150:REM A
1180 DATA  85,105,150,149,105, 86,150,105:REM S
1190 DATA  85,169,150,150,150,150,150,169:REM D
1200 DATA  85,170,149,149,170,149,149,149:REM F
1210 DATA 105,150,149,149,154,150,150,105:REM G
1220 DATA  85,150,150,150,170,150,150,150:REM H
1230 DATA  85,106, 89, 89, 89, 89,153,101:REM J
1240 DATA 150,153,153,165,165,153,153,150:REM K
1250 DATA 105,105,105,105,105, 85, 85,105:REM W
1260 DATA 170,85,106,106,85,106,106,85    :REM E
1270 DATA 170,149,166,166,166,166,166,166:REM T
1280 DATA 170,153,153,153,166,166,166,166:REM Y
1290 DATA 170,105,105,105,105,105,105,150:REM U
1300 DATA 170,150,105,105,105,105,105,150:REM O
1310 DATA 125,119,119,119,117,245,245,245:REM MARK #1
1320 DATA 190,189,187,187,122,250,250,250:REM MARK #2
1330 :
1340 REM  SCREEN PATTERN (BY SCREEN CODE)
1350 DATA 6,2,9,9,2,9,9,2,7,6,2,9,9,2,9,9,2,9,9,2,7,6,2,9
1360 DATA 0,8,9,9,8,9,9,8,1,0,8,9,9,8,9,9,8,9,9,8,1,0,8,9
1370 DATA 0,8,9,9,8,9,9,8,1,0,8,9,9,8,9,9,8,9,9,8,1,0,8,9
1380 DATA 0,8,18,9,8,19,9,8,1,0,8,20,9,8,21,9,8,22,9,8,1,0,8,23
1390 DATA 0,8,9,9,8,9,9,8,1,0,8,9,9,8,9,9,8,9,9,8,1,0,8,9
1400 DATA 0,8,1,0,8,1,0,8,1,0,8,1,0,8,1,0,8,1,0,8,1,0,8,1
1410 DATA 0,8,1,0,8,1,0,8,1,0,8,1,0,8,1,0,8,1,0,8,1,0,8,1
```

```
1420 DATA 0,8,1,0,8,1,0,8,1,0,8,1,0,8,1,0,8,1,0,8,1,0,8,1
1430 DATA 0,10,1,0,11,1,0,12,1,0,13,1,0,14,1,0,15,1,0,16,1,0,17,1
1440 DATA 4,3,5,4,3,5,4,3,5,4,3,5,4,3,5,4,3,5,4,3,5,4,3,5
```

胸キュンなんだこの曲。彼女と2人で聞きたい“サウンド オブ サイレンス”


```
10 REM ***********************************
20 REM *                                 *
30 REM *   COMMODORE - 64                *
40 REM *                                 *
50 REM *   MUSIC - SOUND OF SILENCE      *
60 REM *                                 *
70 REM *      PROGRAM BY  KEIJI.BESSI    *
80 REM *                                 *
90 REM ***********************************
95 :
100 REM **** SYOKI SETTEI *****
110 DIM C1(27,1),C2(27,1),C3(37,1)
120 FOR I=0 TO 27
122 READ CL,CH
124 C1(I,0)=CL:C1(I,1)=CH
126 NEXT
128 :
130 FOR I=0 TO 27
132 READ CL,CH
134 C2(I,0)=CL:C2(I,1)=CH
136 NEXT
138 :
140 FOR I=0 TO 37
142 READ CL,CH
144 C3(I,0)=CL:C3(I,1)=CH
146 NEXT
148 :
160 SID=13*4096+4*256
170 POKE SID+24,15
199 :
200 REM **** ATAK  DYCKY ****
210 POKE SID+ 5,0*16+9
220 POKE SID+12,0*16+9
230 POKE SID+19,0*16+9
235 :
240 REM **** SASUTEAN  RELEASE ****
250 POKE SID+ 6,0*16+10
```

```
260 POKE SID+13,0*16+9
270 POKE SID+20,0*16+11
275 :
280 REM **** PALUSE - LOW ****
290 POKE SID+ 2,255
300 POKE SID+ 9,255
310 POKE SID+16,255
315 :
320 REM **** PALUSE - HI ****
330 POKE SID+ 3,0
340 POKE SID+10,8
350 POKE SID+17,8
355 :
400 REM ***** ENSOU *****
410 READ T1,T2,T3,TM
415 IF T1=-1 THEN RUN
420 G1=1:IF T1=9 THEN G1=0:T1=T4
430 G2=1:IF T2=9 THEN G2=0:T2=T5
440 G3=1:IF T3=9 THEN G3=0:T3=T6
445 :
450 POKE SID,C1(T1,0):POKE SID+1,C1(T1,1)
460 POKE SID+7,C2(T2,0):POKE SID+8,C2(T2,1)
470 POKE SID+14,C3(T3,0):POKE SID+15,C3(T3,1)
475 :
480 POKE SID+4,16+G1:POKE SID+11,16+G2:POKE SID+18,64+G3
485 :
490 FOR I=1 TO TM*55:NEXT
495 :
500 POKE SID+4,16:POKE SID+11,16:POKE SID+18,64
505 T4=T1:T5=T2:T6=T3
510 GOTO 410
998 :
999 REM **** CODE  DATA ****
1000 REM ==== LOW CODE DATA ====
1010 DATA 0,0, 75,5, 156,5, 76,6, 17,7, 125,7, 104,8, 111,9, 0,0, 0,0
1020 DATA 0,0, 151,10, 56,11, 152,12, 35,14, 250,14, 207,16, 222,18, 0,0, 0,0
1030 DATA 0,0, 46,21, 112,22, 48,25, 69,28, 244,29, 159,33, 189,37
1040 :
1050 REM ==== MID CODE DATA ====
1060 DATA 0,0, 104,8, 119,9, 151,10, 56,11, 152,12, 35,14, 250,14, 0,0, 0,0
1070 DATA 0,0, 207,16, 222,18, 46,21, 112,22, 48,25, 69,28, 244,29, 0,0, 0,0
1080 DATA 0,0, 159,33, 189,37, 92,42, 224,44, 95,50, 138,56, 231,59
1090 :
1100 REM ==== HI CODE DATA ====
1110 DATA 0,0, 104,8, 119,9, 151,10, 56,11, 152,12, 35,14, 250,14, 0,0, 0,0
1120 DATA 0,0, 207,16, 222,18, 46,21, 112,22, 48,25, 69,28, 244,29, 0,0, 0,0
1130 DATA 0,0, 159,33, 189,37, 92,42, 224,44, 95,50, 138,56, 231,59, 0,0, 0,0
1140 DATA 0,0, 61,67, 121,75, 183,84, 193,89, 191,100, 21,113, 206,119
1998 :
1999 REM **** MUSIC DATA ****
2000 REM **** 1 ****
2010 DATA 0,0,0,2
2020 DATA 0,12,0,2, 0,16,0,2, 0,13,0,2, 0,16,0,2
2030 DATA 0,12,0,2, 0,16,0,2, 0,13,0,2, 0,16,0,2.5
2040 DATA 0,12,0,2, 0,16,0,2, 17,13,22,2, 9,16,22,2
2050 DATA 22,12,24,2, 9,16,24,2, 24,13,26,2, 9,16,26,2
2060 DATA 23,11,25,2, 9,15,9,2, 9,12,9,2, 16,15,9,2
```

```
2070 DATA 16,11,9,2, 9,15,9,2, 13,12,9,2, 9,15,9,2.2
2080 REM **** 2 ****
2090 DATA 6,11,0,2, 9,15,21,2, 9,12,21,2, 9,15,21,2
2100 DATA 16,11,23,2, 9,15,23,2, 9,12,25,2, 9,15,25,2
2110 DATA 17,12,24,2, 9,16,9,2, 9,13,9,2, 9,16,9,2
2120 DATA 7,12,9,2, 9,16,9,2, 9,13,9,2, 9,16,9,2.2
2130 DATA 12,14,0,2, 9,21,24,2, 9,15,24,2, 9,21,24,2
2140 DATA 22,14,26,2, 9,21,26,2, 9,15,31,2, 9,21,31,2
2150 DATA 15,14,32,2, 9,22,9,2, 9,17,32,2, 9,22,31,2
2160 DATA 12,14,9,2, 9,21,9,2, 9,15,9,2, 9,21,9,2.2
2170 REM **** 3 ****
2180 DATA 12,14,0,2, 9,21,0,2, 9,15,24,2, 9,21,24,2
2190 DATA 22,14,26,2, 9,21,26,2, 9,15,31,2, 9,21,31,2
2200 DATA 15,14,32,2, 9,22,9,2, 9,17,32,2, 9,22,31,2
2210 DATA 12,14,9,2, 9,21,9,2, 9,15,9,2, 9,21,9,2
2220 DATA 12,14,0,2, 9,21,0,2, 9,15,24,2, 9,21,24,2
2230 DATA 15,14,32,2, 9,22,32,2, 9,17,9,2, 9,22,9,2
2240 DATA 15,14,9,2, 9,22,9,2, 9,17,32,2, 9,22,33,2
2250 DATA 5,14,34,2, 9,22,34,2, 9,17,9,2, 9,22,9,2.2
2260 REM **** 4 ****
2270 DATA 15,14,33,2, 9,22,32,2, 9,17,9,2, 9,22,9,2
2280 DATA 12,14,31,2, 9,21,9,2, 9,15,9,2, 9,21,9,2
2290 DATA 12,14,9,2, 9,21,9,2, 9,15,32,2, 9,21,31,2
2300 DATA 22,14,26,2, 9,21,9,2, 9,15,9,2, 9,21,9,2
2310 DATA 16,14,9,2, 9,21,9,2, 9,15,9,2, 9,21,9,2.2
2320 DATA 12,14,0,2, 9,21,24,2, 9,15,24,2, 9,21,24,2
2330 DATA 13,13,31,2, 9,21,9,2, 9,12,9,2, 9,21,9,2.2
2340 DATA 6,13,0,2, 9,21,0,2, 9,12,0,2, 9,21,23,2
2350 REM **** 5 ****
2360 DATA 7,11,24,2, 9,16,22,2, 9,13,9,2, 9,16,9,2
2370 DATA 17,12,9,2, 9,16,9,2, 9,13,9,2, 9,16,9,2.5
2380 DATA 0,12,0,2, 0,16,22,2, 17,13,22,2, 9,16,22,2
2390 DATA 22,12,24,2, 9,16,24,2, 24,13,26,2, 9,16,26,2
2400 DATA 23,11,25,2, 9,15,9,2, 9,12,9,2, 16,15,9,2
2410 DATA 16,11,9,2, 9,15,9,2, 13,12,9,2, 9,15,9,2.2
2420 DATA 6,11,0,2, 9,15,0,2, 9,12,21,2, 9,15,21,2
2430 DATA 16,11,23,2, 9,15,23,2, 9,12,25,2, 9,15,25,2
2440 REM **** 6 ****
2450 DATA 17,12,24,2, 9,16,9,2, 9,13,9,2, 9,16,9,2
2460 DATA 7,12,9,2, 9,16,9,2, 9,13,9,2, 9,16,9,2.2
2470 DATA 12,14,0,2, 9,21,0,2, 9,15,24,2, 9,21,24,2
2480 DATA 22,14,26,2, 9,21,26,2, 9,15,31,2, 9,21,31,2
2490 DATA 15,14,32,2, 9,22,9,2, 9,17,32,2, 9,22,31,2
2500 DATA 12,14,9,2, 9,21,9,2, 9,16,9,2, 9,21,9,2.2
2510 DATA 12,14,0,2, 9,21,24,2, 9,15,24,2, 9,21,24,2
2520 DATA 22,14,26,2, 9,21,26,2, 9,15,31,2, 9,21,31,2
2530 REM **** 7 ****
2540 DATA 15,14,32,2, 9,22,9,2, 9,17,32,2, 9,22,31,2
2550 DATA 12,14,9,2, 9,21,9,2, 9,16,9,2, 9,21,9,2.3
2560 DATA 15,14,0,2, 9,22,0,2, 9,17,24,2, 9,22,24,2
2570 DATA 5,14,32,2, 9,22,9,2, 9,17,32,2, 9,22,32,2
2580 DATA 15,14,9,2, 9,22,9,2, 9,17,32,2, 9,22,33,2
2590 DATA 5,14,34,2, 9,22,9,2, 9,17,34,2, 9,22,34,2
2600 DATA 15,14,33,2, 9,22,32,2, 9,17,9,2, 9,22,31,2
2610 DATA 12,14,9,2, 9,21,9,2, 9,15,9,2, 9,21,9,2.2
2620 REM **** 8 ****
2630 DATA 22,14,0,2, 9,21,31,2, 9,15,32,2, 9,21,31,2
```

```
2640 DATA 12,14,26,2, 9,21,9,2, 9,15,9,2, 9,21,9,2
2650 DATA 17,12,9,2, 9,16,9,2, 9,13,9,2, 9,16,9,2.2
2660 DATA 7,12,0,2, 9,16,24,2, 9,13,24,2, 9,16,24,2
2670 DATA 6,13,31,2, 9,21,9,2, 9,13,9,2, 9,21,9,2.2
2680 DATA 16,13,0,2, 9,21,0,2, 9,13,0,2, 9,21,23,2
2690 DATA 17,12,24,2, 9,16,22,2, 9,13,9,2, 9,16,9,2
2700 DATA 7,12,9,2, 9,16,9,2, 9,13,9,2, 9,16,9,2.7
2710 ***** NO.1 END ****
2720 **** N0.2 ****
2730 DATA 0,12,0,2, 0,16,22,2, 17,13,22,2, 9,16,22,2
2740 DATA 22,12,24,2, 9,16,24,2, 24,13,26,2, 9,16,26,2
2750 DATA 23,11,25,2, 9,15,9,2, 9,12,9,2, 16,15,9,2
2760 DATA 16,11,9,2, 9,15,9,2, 13,12,9,2, 9,15,9,2.2
2770 DATA 6,11,0,2, 9,15,0,2, 9,12,21,2, 9,15,21,2
2780 DATA 16,11,23,2, 9,15,23,2, 9,12,25,2, 9,15,25,2
2790 REM **** 6 ****
2800 DATA 17,12,24,2, 9,16,9,2, 9,13,9,2, 9,16,9,2
2810 DATA 7,12,9,2, 9,16,9,2, 9,13,9,2, 9,16,9,2.2
2820 DATA 12,14,0,2, 9,21,0,2, 9,15,24,2, 9,21,24,2
2830 DATA 22,14,26,2, 9,21,26,2, 9,15,31,2, 9,21,31,2
2840 DATA 15,14,32,2, 9,22,9,2, 9,17,32,2, 9,22,31,2
2850 DATA 12,14,9,2, 9,21,9,2, 9,16,9,2, 9,21,9,2.2
2860 DATA 12,14,0,2, 9,21,24,2, 9,15,24,2, 9,21,24,2
2870 DATA 22,14,26,2, 9,21,26,2, 9,15,31,2, 9,21,31,2
2880 REM **** 7 ****
2890 DATA 15,14,32,2, 9,22,9,2, 9,17,32,2, 9,22,31,2
2900 DATA 12,14,9,2, 9,21,9,2, 9,16,9,2, 9,21,9,2.3
2910 DATA 15,14,0,2, 9,22,0,2, 9,17,24,2, 9,22,24,2
2920 DATA 5,14,32,2, 9,22,9,2, 9,17,32,2, 9,22,32,2
2930 DATA 15,14,9,2, 9,22,9,2, 9,17,32,2, 9,22,33,2
2940 DATA 5,14,34,2, 9,22,9,2, 9,17,34,2, 9,22,34,2
2950 DATA 15,14,33,2, 9,22,32,2, 9,17,9,2, 9,22,31,2
2960 DATA 12,14,9,2, 9,21,9,2, 9,15,9,2, 9,21,9,2.2
2970 REM **** 8 ****
2980 DATA 22,14,0,2, 9,21,31,2, 9,15,32,2, 9,21,31,2
2990 DATA 12,14,26,2, 9,21,9,2, 9,15,9,2, 9,21,9,2
3000 DATA 17,12,9,2, 9,16,9,2, 9,13,9,2, 9,16,9,2.2
3010 DATA 7,12,0,2, 9,16,24,2, 9,13,24,2, 9,16,24,2
3020 REM **** 9 ****
3030 DATA 6,13,31,2, 9,21,9,2, 9,13,9,2, 9,21,9,2.2
3040 DATA 16,13,0,2, 9,21,0,2, 9,13,0,2, 9,21,23,2
3050 DATA 17,12,24,2, 9,16,22,2, 9,13,9,2, 9,16,9,2
3060 DATA 7,12,9,2, 9,16,9,2, 9,13,9,2, 9,16,9,2
3070 DATA 17,12,0,2, 9,16,0,2, 9,13,0,2, 9,16,0,2
3080 DATA 7,12,0,2, 9,16,0,2, 9,13,0,2, 9,16,0,2
3090 DATA 7,16,22,2, 9,9,9,10
3100 REM ***** NO.2 END *****
3110 DATA -1,1,1,1
3120 REM ******** END *********
```

高尚なメロディに身をゆだねる、たまにはネ。“ガボット(バッハ)”

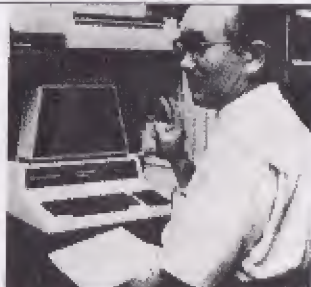

```
5 REM  ********************************
10 REM *                              *
20 REM *   COMMODORE-64  MUSICIAN     *
30 REM *                              *
40 REM *    MUSIC - GABOOTO (BACH)    *
50 REM *                              *
60 REM *    PROGRAM BY  K.B   58.7.11 *
70 REM *                              *
80 REM ********************************
90 :
100 REM **** SYOKI SETTEI ****
110 DIM TL(77,1),TH(77,1),HL(8),HH(8)
120 SID=13*4096+4*256
130 POKE SID+24,15
140 :
150 REM **** CODE DATA READ - L ****
160 FOR I=0 TO 77
170 READ CL,CH
180 TL(I,0)=CL:TL(I,1)=CH
190 NEXT
200 :
210 REM **** CODE DATA READ - H ****
220 FOR I=0 TO 77
230 READ CL,CH
240 TH(I,0)=CL:TH(I,1)=CH
250 NEXT
260 :
300 REM **** ATACK - DICKEY ****
310 POKE SID+5, 0*16+9
320 POKE SID+12,0*16+9
330 POKE SID+19,0*16+9
340 :
350 REM **** SASUTEEN - RELEASE ****
360 POKE SID+6, 0*16+10
370 POKE SID+13,0*16+11
380 POKE SID+20,0*16+11
390 :
400 REM **** PALUSE - LOW ****
410 POKE SID+2, 255
420 POKE SID+9, 255
430 POKE SID+16,255
```

```
440 :
450 REM **** PALUSE - H I ****
460 POKE SID+3, 8
470 POKE SID+10,8
480 POKE SID+17,8
490 :
500 REM **** CODE HENKAN ****
510 GOSUB 1000
520 :
600 REM **** ENSOU ****
610 READ T1,T2,T3,TM
620 IF T1=99 THEN RUN
630 IF T1<0 THEN T1=ABS(T1)+40
640 IF T2<0 THEN T2=ABS(T2)+40
650 IF T3<0 THEN T3=ABS(T3)+40
660 G1=1:IF T1=9 THEN G1=0:T1=E1
670 G2=1:IF T2=9 THEN G2=0:T2=E2
680 G3=1:IF T3=9 THEN G3=0:T3=E3
690 POKE SID,TL(T1,0):POKESID+1,TL(T1,1)
700 POKE SID+7,TH(T2,0):POKE SID+8,TH(T2,1)
710 POKE SID+14,TH(T3,0):POKE SID+15,TH(T3,1)
720 POKE SID+4,64+G1:POKE SID+11,64+G2:POKE SID+18,64+G3
730 FOR I=1 TO 50*TM:NEXT
740 E1=T1:E2=T2:E3=T3
750 POKE SID+4,64:POKE SID+11,64:POKE SID+18,64
760 GOTO 610
770 :
1000 REM **** CODE HENKAN SUB ****
1010 REM"┌──────────────────────────────────┐"
1020 REM"| HENKAN CODE (HC)   NO-0          |"
1030 REM"|                                  |"
1040 REM"| SHARP  1-1  2-2  3-3  4-4        |"
1050 REM"|                                  |"
1060 REM"| FLAT   1-5  2-6  3-7  4-8        |"
1070 REM"└──────────────────────────────────┘"
1080 HC=1
1090 :
1100 REM **** HENKAN DATA READ ****
1110 FOR I=0 TO 8
1120 READ HD
1130 HL(I)=HD
1140 NEXT
1150 :
1160 FOR I=0 TO 8
1170 READ HD
1180 HH(I)=HD
1190 NEXT :IF HC=0 THEN RETURN
1200 :
1210 REM **** SHARP-FLAT  HANBETU ****
1220 IF HC>4 THEN 1500
1290 :
1300 REM **** HENKAN  SHARP-TL ****
1310 FOR I=1 TO HC
1320 FOR J=0 TO 3
1330 CL=TL(HL(I)+10*J,0):CH=TL(HL(I)+10*J,1)
1340 TL(HL(I)+10*J,0)=TL(HL(I)+40+10*J,0)
1350 TL(HL(I)+10*J,1)=TL(HL(I)+40+10*J,1)
```

```
1360 TL(HL(I)+40+10*J,0)=CL
1370 TL(HL(I)+40+10*J,1)=CH
1380 NEXT J:NEXT I
1390 :
1400 REM **** HENKAN  SHARP-TH ****
1410 FOR I=1 TO HC
1420 FOR J=0 TO 3
1430 CL=TH(HH(I)+10*J,0):CH=TH(HH(I)+10*J,1)
1440 TH(HH(I)+10*J,0)=TH(HH(I)+40+10*J,0)
1450 TH(HH(I)+10*J,1)=TH(HH(I)+40+10*J,1)
1460 TH(HH(I)+40+10*J,0)=CL
1470 TH(HH(I)+40+10*J,1)=CH
1480 NEXT J:NEXT I:RETURN
1490 :
1500 REM **** HENKAN  FLAT-TL ****
1510 FOR I=5 TO HC
1520 FOR J=0 TO 3
1530 CL=TL(HL(I)+10*J,0):CH=TL(HL(I)+10*J,1)
1540 TL(HL(I)+10*J,0)=TL(HL(I)+39+10*J,0)
1550 TL(HL(I)+10*J,1)=TL(HL(I)+39+10*J,1)
1560 TL(HL(I)+40+10*J,0)=CL
1570 TL(HL(I)+40+10*J,1)=CH
1580 NEXT J:NEXT I
1590 :
1600 REM **** HENKAN  FLAT-TH ****
1610 FOR I=5 TO HC
1620 FOR J=0 TO 3
1630 CL=TH(HH(I)+10*J,0):CH=TH(HH(I)+10*J,1)
1640 TH(HH(I)+10*J,0)=TH(HH(I)+39+10*J,0)
1650 TH(HH(I)+10*J,1)=TH(HH(I)+39+10*J,1)
1660 TH(HH(I)+40+10*J,0)=CL
1670 TH(HH(I)+40+10*J,1)=CH
1680 NEXT J:NEXT I:RETURN
1690 :
2000 REM **** CODE DATA - LOW ****
2010 DATA 0,0, 166,2, 206,2, 38,3, 137,3, 247,3, 52,4, 184,4, 0,0, 0,0
2020 DATA 0,0, 75,5, 156,5, 76,6, 17,7, 239,7, 104,8, 111,9, 0,0, 0,0
2030 DATA 0,0, 151,10, 56,11, 152,12, 35,14, 222,15, 207,16, 222,18, 0,0, 0,0
2040 DATA 0,0, 46,21, 112,22, 48,25, 69,28, 188,31, 159,33, 189,37, 92,42, 0,0
2050 DATA 0,0, 0,0, 249,2, 86,3, 190,3, 0,0, 116,4, 255,4, 0,0, 0,0
2060 DATA 0,0, 0,0, 241,5, 172,6, 125,7, 0,0, 232,8, 255,9, 0,0, 0,0
2070 DATA 0,0, 0,0, 227,11, 88,13, 250,14, 0,0, 207,17, 253,19, 0,0, 0,0
2080 DATA 0,0, 0,0, 198,23, 175,26, 244,29, 0,0, 158,35, 251,39
2090 :
2100 REM **** CODE DATA - H I ****
2110 DATA 0,0, 104,8, 111,9, 151,10, 56,11, 152,12, 35,14, 222,15, 0,0, 0,0
2120 DATA 0,0, 207,16, 222,18, 46,21, 112,22, 48,25, 69,28, 188,31, 0,0, 0,0
2130 DATA 0,0, 159,33, 189,37, 92,42, 224,44, 95,50, 138,56, 119,63, 0,0, 0,0
2140 DATA 0,0, 61,67, 121,75, 183,84, 193,89, 191,100, 21,113, 238,126, 0,0, 0
2150 DATA 0,0, 232,8, 255,9, 0,0, 227,11, 88,13, 250,14, 0,0, 0,0, 0,0
2160 DATA 0,0, 207,17, 253,19, 0,0, 198,23, 175,26, 244,29, 0,0, 0,0, 0,0
2170 DATA 0,0, 158,35, 251,39, 0,0, 140,47, 94,53, 231,59, 0,0, 0,0, 0,0
2180 DATA 0,0, 61,71, 246,69, 0,0, 23,95, 188,106, 206,119, 0,0
2190 :
2200 REM **** HENKAN DATA ****
2210 DATA 0,2,6,3,7,5,1,4,7
2220 DATA 0,4,1,5,2,7,3,6,2
```

```
2990 :
3000 REM **** MUSIC DATA - GABOOTO ****
3010 DATA 0,0,0,2 : REM *** 1 ***
3020 DATA 23,22,27,2, 33,17,25,2
3030 DATA 32,16,22,2, 9,9,9,2, 22,9,23,2, 9,9,24,2
3040 DATA 21,17,25,2, 9,9,9,2, 31,15,23,2, 9,9,9,2
3050 DATA 27,14,17,2, 9,9,9,2, 17,9,9,2, 9,9,9,2
3060 DATA 16,15,23,2, 9,9,9,2, 26,16,21,2, 9,9,9,2.3
3070 DATA 27,14,16,2, 9,9,9,2, 24,0,17,2, 9,9,21,2
3080 DATA 22,16,22,2, 9,9,17,2, 17,0,21,2, 9,9,16,2
3090 DATA 23,15,21,2, 9,9,17,2, 17,0,16,2, 9,9,15,2
3100 DATA 13,22,27,2, 17,9,9,2, 21,17,25,2, 22,9,9,2
3110 REM *** 2 ***
3120 DATA 23,-21,23,2, 24,9,9,2, 23,-21,23,2, 22,22,24,2
3130 DATA 21,23,25,2, 17,9,9,2, 21,-21,23,2, 22,9,9,2.3
3140 DATA 23,16,-21,2, 24,9,9,2, 25,9,22,2, -26,9,23,2
3150 DATA 27,9,24,2, 9,9,22,2, -26,9,25,2, 9,9,23,2
3160 DATA 27,9,24,2, 9,9,22,2, 22,22,26,2, 23,9,9,2
3170 DATA 24,9,25,2, 9,9,24,2, 14,-21,23,2, 9,9,24,2
3180 DATA 17,0,22,2, 9,9,9,2, 9,9,9,2, 9,9,9,2
3190 DATA 27,14,16,2, 24,16,22,2
3200 DATA 23,22,24,2, 22,9,9,2, 21,9,9,2, 17,9,9,2
3210 DATA 24,0,25,2, 9,9,24,2, 26,9,23,2, 9,9,22,2
3220 REM *** 3 ***
3230 DATA 25,22,25,2, 26,9,9,2, 25,0,22,2, 24,9,9,2
3240 DATA 23,25,27,2, 24,9,9,2, 25,17,25,2, 23,9,9,2
3250 DATA 26,15,23,2, 27,9,9,2, 26,16,24,2, 25,17,25,2
3260 DATA 24,21,26,2, 25,9,9,2, 26,16,24,2, 24,9,9,2
3270 DATA 25,14,-22,2, 9,9,9,2, 15,9,9,2, 9,9,9,2.3
3280 DATA 0,0,17,2, 9,9,-21,2, 9,14,22,2, 9,9,23,2
3290 DATA 9,-12,24,2, 9,9,25,2, 9,12,24,2, 9,13,23,2
3300 DATA 9,14,-22,2, 9,9,23,2, 9,-12,24,2, 9,9,25,2
3310 DATA 25,0,26,2, 9,9,27,2, 25,9,26,2, -26,9,25,2
3320 DATA -27,9,24,2, 9,9,25,2, 25,9,26,2, 9,9,27,2
3330 REM *** 4 ***
3340 DATA 23,0,31,2, 22,9,26,2, 23,9,27,2, 24,9,9,2
3350 DATA 25,15,23,2, 9,9,9,2, 15,16,24,2, 9,9,-22,2
3360 DATA 21,15,23,2, 9,9,9,2, 15,9,9,2, 9,9,9,2.3
3370 DATA 11,15,17,2, 12,9,9,2, 13,17,23,2, 14,9,9,2
3380 DATA 15,22,25,2, 16,9,9,2, 17,9,9,2, 21,9,9,2
3390 DATA -22,0,21,2, 23,9,17,2, 24,9,16,2, 25,9,15,2
3400 DATA 26,9,-24,2, 27,9,22,2, 26,9,23,2, 25,9,9,2
3410 DATA 24,9,14,2, 23,9,15,2, 22,9,16,2, 21,9,17,2
3420 DATA 17,9,21,2, 16,9,16,2, 15,9,17,2, 14,9,21,2
3430 DATA 13,9,22,2, 14,9,21,2, 15,9,22,2, 16,9,23,2
3440 REM *** 5 ***
3450 DATA 17,14,16,2, 14,9,9,2, 12,9,9,2, 14,9,9,2.3
3460 DATA 7,26,31,2, 11,9,9,2, 12,22,26,2, 13,9,9,2
3470 DATA 14,21,24,2, 15,9,9,2, 14,9,24,2, 13,9,25,2
3480 DATA 12,21,26,2, 13,9,9,2, 14,16,24,2, 15,9,9,2
3490 DATA 16,14,22,2, 17,9,9,2, 16,9,22,2, 15,9,23,2
3500 DATA 14,21,24,2, 15,9,25,2, 16,9,26,2, 17,9,24,2
3510 DATA 21,17,25,2, 9,9,23,2, 16,21,26,2, 9,9,9,2
3520 DATA 17,16,24,2, 9,9,9,2, 7,9,9,2, 9,9,25,2
3530 DATA 13,17,25,2, 9,9,9,2, 9,9,9,2, 9,9,9,2
3540 REM *** NO-1 END ***
3550 REM *** 2 ***
```

```
3560 DATA 27,14,16,2, 24,16,22,2
3570 DATA 23,22,24,2, 22,9,9,2, 21,9,9,2, 17,9,9,2
3580 DATA 24,0,25,2, 9,9,24,2, 26,9,23,2, 9,9,22,2
3590 REM *** 3 ***
3600 DATA 25,22,25,2, 26,9,9,2, 25,0,22,2, 24,9,9,2
3610 DATA 23,25,27,2, 24,9,9,2, 25,17,25,2, 23,9,9,2
3620 DATA 26,15,23,2, 27,9,9,2, 26,16,24,2, 25,17,25,2
3630 DATA 24,21,26,2, 25,9,9,2, 26,16,24,2, 24,9,9,2
3640 DATA 25,14,-22,2, 9,9,9,2, 15,9,9,2, 9,9,9,2.3
3650 DATA 0,0,17,2, 9,9,-21,2, 9,14,22,2, 9,9,23,2
3660 DATA 9,-12,24,2, 9,9,25,2, 9,12,24,2, 9,13,23,2
3670 DATA 9,14,-22,2, 9,9,23,2, 9,-12,24,2, 9,9,25,2
3680 DATA 25,0,26,2, 9,9,27,2, 25,9,26,2, -26,9,25,2
3690 DATA -27,9,24,2, 9,9,25,2, 25,9,26,2, 9,9,27,2
3700 REM *** 4 ***
3710 DATA 23,0,31,2, 22,9,26,2, 23,9,27,2, 24,9,9,2
3720 DATA 25,15,23,2, 9,9,9,2, 15,16,24,2, 9,9,-22,2
3730 DATA 21,15,23,2, 9,9,9,2, 15,9,9,2, 9,9,9,2.3
3740 DATA 11,15,17,2, 12,9,9,2, 13,17,23,2, 14,9,9,2
3750 DATA 15,22,25,2, 16,9,9,2, 17,9,9,2, 21,9,9,2
3760 DATA -22,0,21,2, 23,9,17,2, 24,9,16,2, 25,9,15,2
3770 DATA 26,9,-24,2, 27,9,22,2, 26,9,23,2, 25,9,9,2
3780 DATA 24,9,14,2, 23,9,15,2, 22,9,16,2, 21,9,17,2
3790 DATA 17,9,21,2, 16,9,16,2, 15,9,17,2, 14,9,21,2
3800 DATA 13,9,22,2, 14,9,21,2, 15,9,22,2, 16,9,23,2
3810 REM *** 5 ***
3820 DATA 17,14,16,2, 14,9,9,2, 12,9,9,2, 14,9,9,2.3
3830 DATA 7,26,31,2, 11,9,9,2, 12,22,26,2, 13,9,9,2
3840 DATA 14,21,24,2, 15,9,9,2, 14,9,24,2, 13,9,25,2
3850 DATA 12,21,26,2, 13,9,9,2, 14,16,24,2, 15,9,9,2
3860 DATA 16,14,22,2, 17,9,9,2, 16,9,22,2, 15,9,23,2
3870 DATA 14,21,24,2, 15,9,25,2, 16,9,26,2, 17,9,24,2
3880 DATA 21,17,25,2, 9,9,23,2, 16,21,26,2, 9,9,9,2
3890 DATA 17,16,24,2, 9,9,9,2, 7,9,9,2, 9,9,25,2
3900 DATA 13,17,25,2, 9,9,9,2, 9,9,9,2, 9,9,9,8
3910 REM *** NO-2 END ***
3920 DATA 99,0,0,0:REM ** REPEAT **
```

高倉杏二郎▶
自営業

CBM3032でみっちりお勉強の1カ月。

ヤッタゾ！ 町内でも評判のパソコンフリーク、助監督は50才。

「野球にパソコンを導入したのは、評論家の野村さんより早い」と胸をはるのは、東京の八王子市にある〝タカクラメガネ〟のご主人、高倉杏二郎さん。

高倉さんは草野球チーム〝八幡町フェアーズ〟の助監督。チーム結成当時は現役の1塁手だったが、翌年に引退し、めでたく助監督のポストに納まった。しかし「肩書きはりっぱでも、実際はスコアブックとにらめっこしてるだけ」だったという。

あるとき高倉さんは、ブームになりかけていたパソコンで、スコア・データの整理をしようと思いたった。生来の機械好きも手伝ってのことだ。

「あなたのトシでパソコンをマスターしようとする人は少ない」という周囲の励ましともひやかしともつかない声をよそに、コモドールCBM3032のフルシステムを買いこみ、コンピュータ教室で猛勉強した。そして1カ月余りで、お年玉くじ付き年賀ハガキの、当落を調べるプログラムを作るまでになった。

「今思えば、ほんの子どもだましの単純なもの」だったそうだが、作った当初はうれしくてたまらなかったという。

パソコンの操作にも慣れ、いよいよ〝フェアーズ〟用のソフト作りに取り組むようになる。さすがメガネ屋さんだけあって、目が悪くならないようにと、特製のフィルターをブラウン管に被せ、それでも目をしょぼつかせながら、夜ふけの2時、3時までパソコンに向かった。初めは高倉さんの健康を気づかっていた家族も、しまいにはあきらめるほどの熱の入れようだった。

苦心さんたんのすえ、作り上げたプログラムは、スコア・データと個人成績が一瞬にして画面に表われ、おまけに出席率まではじき出される。だから選手によって評判もまちまち。成績のよい選手は、まだデータ入力中の高倉さんをつかまえて、せかすことしきりなのだが──。

かくして、開幕直後の5連戦、3勝2敗、貯金1、平均年齢36・5歳という〝八幡町フェアーズ〟にとって、高倉さんのパソコンは必勝の武器(?)となっているのである。

野球のみならず、町内ゴルフ・コンペや、本業のほうでも仕入れ管理などにパソコンを駆使、今や町内でも有名なパソコン・フリークというわけなのである。

ちょっと拝見コモドール PART 1

VIC·Report

NHKの「600こちら情報部」でおなじみ、帯淳子さんがMAXで作曲に挑戦‼

音色自由自在。作曲コンピュータでつくる、迷曲⁉

「こんにちは！」、ドアの向こうから、大きなくりくり目玉の帯淳子さんの顔が現れた。NHKテレビ「600こちら情報部」でおなじみの笑顔である。今日は帯淳子さんにコモドールのマックスマシーンにトライしてもらうために、機械を持ってお宅を訪問したというわけ。

淳子さん、番組では、子供たちの大好きなコンピュータの話題を取り上げている。ロバート・エーブルのコンピュータ・グラフィックの質感に驚き、昨年のお正月にはパソコン制御の未来ハウスもスタジオにつくったという。しかし、自分でキーボードを叩くことはほとんど初めてとのこと。

マックスは、表示に家庭用テレビを使うので、本体のキーボード、ACアダプター、それにキーボードと家庭用テレビをつなぐスイッチボックスの3つに分かれている。

まず、居間にあるテレビにスイッチボックスを接続しようとしたのだが、混合同軸系のためにうまくいかず、2階からお母さんのテレビを持ってくる。これもVHFアンテナの所へつなぐのだが、画像が乱れる。垂直、水平のレベルを合わせるが、色ずれがひどく、画面が読み取りにくい。どうやら古いテレビは苦手の機械らしい。

3台目、お手伝いさんのテレビでやっとバッチリの絵が出る。これで安心。さて本番、ミュージックカートリッジを入れてみる。このあたりマニュアルに沿って慎重に、淳子さんのくりくり目が真剣になる。そして、スイッチオン。画面にタイトル、何でもいいからキーを押せという指令が出る。マックスのキーボードは、1枚型のため、普通のキーボードを押し慣れた人にはやや感じがつかみにくい。試しにBのキーを押してみる。

いよいよ音楽機能の選択画面である。0か

キーボードを鍵盤のように使う。半音も出すことができる。

◀ピアノとコンピュータ、さて、どっちが名曲を生み出すか。

音を出すと即座に音符が画面に現れる。便利だ。

ら７までの数字から選べばよい。(0)チューズインストルメンツ、(1)プレイサンプル、(2)キーボードピアノ、(3)セットエフェクト、(4)エディットソング、(5)プレイソング、(6)ロードソング、(7)セーブソング。

まずφのキーを押し、楽器の選択をする。アコーデオン、カリオペ、リズムサウンド、フルート、ハープシコード、オルガン、ハーモニカ、シロホン、バンジョーの９つの楽器と、スペシャルと称する楽器のうちから３つ楽器を選択することができる。淳子さんは、まず鍵盤楽器からハープシコード、管楽器からフルート、最後に弦楽器からバンジョーを選んだ。また、画面は、選択画面に戻る。今度は、プレイサウンドを押してみる。

妙なるバッハの旋律とともに音符が画面を流れていく。コンピュータが自動演奏、人間思わず拍手！

とどうだ、先ほどの編成でサンプル曲が演奏されてくる。画面には、音符が流れていく。思わず拍手。曲はバッハの「主よ、人の望みの喜びよ」である。

この辺で淳子さんは、マシーンに熱中してきた。「イヨッ」とか「キャッ」とか言いながら、楽器の編成を変えては、このサンプル曲を演奏させている。編成の変化によって曲の表情がさまざまに変わっていくのが面白い。楽器もそっくりとはいかないがイメージは出ている。「シロホンとハープシコードはいけますね」と淳子さん。

次は、キーボード、ピアノに挑戦、これは２のキーを押す。淳子さんは大学の声楽科に籍を置き、レコードを出したこともある。音楽にはちょっとうるさいほうなのだ。

キーボードのGから横並びのキー８つで１オクターブの音が出る。音符の長さは、左隅のキーWA……等で決める。たとえばWは全音符であり、Aは２分音符である。さらに、＋（プラス）、－（マイナス）のキーでオクターブが変わる。全部で９オクターブの音が出るということなので、キーボードを鍵盤のように使えば、右手と左手で自由に演奏することができる。

淳子さんは最初のうちは手探りで音を探していたが、何やら曲を弾きだした。さだまさし、イルカ、聖子ちゃんと聞き慣れた曲が部屋の中を駆けめぐる。少しキーボードのタッチの感じが悪いのを除けば、楽器のできる人は結構楽しめる。

シンセサイザーの機能を持っているセットエフェクトは難しそうなので飛ばす。４のキーを押しエディットソングに向かう。これは作曲に使えるという触れ込み。

淳子さんはマニュアルと首っ引きで参照、先ほどよりやや難しいという。そこでサンプル、あみんの「待つわ」を入力してみる。入力はベーシックと同じ。ドレミをCDEで、それにオクターブと音符の長さを指示し行番号をつけて入れていく。

音符の入力に悪戦苦闘、インスピレーション雲散霧消。作曲マシンの使いこなしには、デジタル思考が必要だ。

たとえば、0010　V305IEDECDCO4B……

ここで悪戦苦闘となった。キーの打ち間違えや読み違いなどで、８小節を入れるのに３分もかかってしまう。根気と集中力の勝負だ。この打込みさえ我慢できれば、楽器の編成など自由自在、テープに取っておけばカラオケにもなる。

さて、肝心の作曲。淳子さん、「待つわ」のプログラムを手本に、行番号と音符を即興に打ち込んでみた。８小節ほど打ち込んで、バンジョーとフルートとリズムサウンドを使って曲を流してみる。うーん、やはり名曲とは言えませんなー。

作曲コンピュータといっても、コンピュータが曲をつくるわけではない。考えるのはあくまで人間。音符を記憶し、演奏するマシンと考えたほうがいい。五線譜を書く代わりに、記号でデータを打ち込むわけだ。それが自動演奏につながるのだから、まごうことなき作曲マシンと言えそうだが、実際に曲をつくるのは大変難しい。音符の入力をマスターするのに時間がかかるからだ。ピアノを弾いて、五線譜に書いたほうがずっと早い。

一通りやった感想を淳子さんに聞くと、「使いはじめると夢中になるわね。作曲は難しいけど、楽器としては面白いわ。次は、もっと勉強してシンセサイザーに挑戦……」

とのこと。３万４８００円という値段を考えると、入門用としては多彩な機能を持った手軽な機械だと思える。

VIC・Report

ちょっと拝見コモドール PART 2

麗(うるわ)しのギャルがいま、夢中！

ギャルたちはのろまな男を好まない。デジタル感覚でスピーディーに応えてくれる相手を求めている。ここにレポートした4人のギャルの恋人はパソコン。彼女たちのどんな興味にも、ヴィジュアルで、正確に反応するからだ。もちろんパソコンに遠慮や気おくれなんかしていない。実用でも遊びでも、モニターと見つめ合うのはイマジネイティヴで胸キュンもの！　ゲームといえばパチンコを連想するキミ、とくと読むべし。

ミニ・ステーションでもDJの データ・ファイルは大放送局なみ。 先端を行くには身軽が一番

キミは、ミニFM放送局って知っているかな。日本の法律では、半径500メートル位しか届かない微弱な電波なら、誰でも自由に放送してよいことになっている。レコードをかけたり、おしゃべりをしたり……ミニFM放送局は、いまひそかなブームになろうとしている。

鎌滝佐知子クンも、いま東京のとある場所から、日夜ディスク・ジョッキーを流している。

ところが、ここからが他のミニFM放送局と一味も二味も違うのだ。

その秘密は、ミキサーと並んでいるコモドールMAXマシーンだ。このパソコンには、佐知子クンが２ヶ月もかかって作ったという、オリジナル・プログラムがインプットしてある。

そのメニューをちょっと紹介すると、まず、『ビルボードTOP100。』キーボードから曲目をインプットすると、たちどころにエントリーした週から現在までのデータが画面に表示されるというスグレモノ。

また、1958年5月第1週のトップテンなんてのもOK。とにかく、アメリカンTOP40顔負けの、知識の宝庫なのだ。

こんなことで驚いてはいけないゾ。彼女のD・Jには、MAXマシーンで作ったコンピュータ・ミュージックが流れるのだ。現在、約40曲ストックがあり、コールサインのバックや番組のイントロ、ジングルなど、それぞれ曲が変わるというからスゴイ。

「いま計画しているのは、このMAXマシーンで、YMOみたいにシンセサイザーをコントロールすること。そのためには、もっとパソコンのこと知らなくっちゃいけないんだけど……ルックスと腕に自信のある彼、募集中です。」音楽大好き、20才。

MAXマシーン￥３４,８００

半径500m以内に彼が？　鎌滝佐知子クン(聖心女大２年)

若い才能が脚光を浴びる世界。 いまやそれはロックではなく、 パソコン・ソフトの世界だ。

「２日もキーボードにさわらないと、もう指がウズウズしちゃって、いても立ってもいられないの」というから、ほとんど病気を通り越して、真性テクノ・フィーバーに間違いないのは、鈴木央子(ふみこ)クン。

おかげで央子クンの自宅のテレビはいまや番組受像機の機能をほとんど停止し、MAXマシーン専用ディスプレイと化しつつある。

テレビ・ゲーム歴はテニス・ゲーム以来というから、もうかれこれ××年……。でも正真正銘の18才ということは、要するに早熟なのであります。

かっては、ゲーム・センターに通うことしきりだったそうだが、ある日、ボーイ・フレンドにパソコンというものがあるのを教えられ、彼女は小踊りせんばかりに喜こんだ。

「だって、自分だけの専用ゲーム・マシーンが持てるなんて、夢のような話だったんだモン」。

以来、央子クンはセッセ、セッセとお小遣いをため、このたびめでたくMAXマシーンを手に入れた。

「カートリッジをポンとさしこむだけで、プログラムがスタートするから、カセットみたいに何分もじっとガマンしなくてもよいし、画面はカラフルだし、おまけにサウンドがまたスグレているのよね」と央子クン。

そしていま、彼女は自作のゲーム・プログラムに挑戦している。BASICだから、カートリッジのようなスピード感を出すのは大変だけど、逆にじっくりと時間をかけてパソコンと人間が頭脳で対決するとか。画面、音楽、プログラムすべて自作のゲームは、近く公開する予定だ。

「私の夢は、パソコンのゲーム・デザイナーになることなの。アメリカでは、ゲーム作りのスーパー・スターがもう何人も生まれているのヨ。まるでロック・アーティストみたいに、ファンも大勢いて……。発売の予告が出ただけで、ドッと注文が殺倒するんだって。そんなふうになれたら、私、最高にシアワセよ」。

MAXマシーン￥３４,８００

わたしの発想についてこられない表現方法は芸術的じゃない。グラフィックにはパソコン!

デザイン事務所でアルバイト中の、大塚美和クンは、目下コンピュータ・グラフィックスに夢中になっている。

キーボードから、何やらプログラムをインプットすると、画面に、赤、緑、ピンク、黄色……いろんな色の図形が次から次へと、出ては消え、出ては消える。また何やらインプットすると、今度は、出た図形の中に色が塗られる。そのリズムが何とも面白いのだ。

プログラムの勉強を始めてから、まだ半年足らずというのに、まるで魔法使いのように、器用にパソコンを操っている。美和クンの動かしているのはカラー・パーソナル・コンピュータ、コモドール64だ。

「ネオンサインとか、レーザー光線とか、発光体ってとってもポップな感じするの」と彼女は言う。「パソコンのディスプレイも同じね。コンピュータ・グラフィックスは、ポップ・アートのニュー・ウェイブってとこかしら……」なんて、わかったような、わからないような……。ボクだって、アンディ・ウォーホルくらい知っているぞ!!

何はともあれ、楽しいのだからよい。「どうして、コンピュータ・グラフィックスに興味を持ったの?」

「高校の頃は、油絵を描いたり、ときどきアニメーションなんか作って、文化祭のときに発表していたのね。それでパソコンに、実際にふれたのは、この事務所にアルバイトに来てからなんだけど。ディスプレイに描いた絵を見て、一瞬、ピンときちゃったんだ。コレヲツカエバ新シイアニメが作レルゾ。

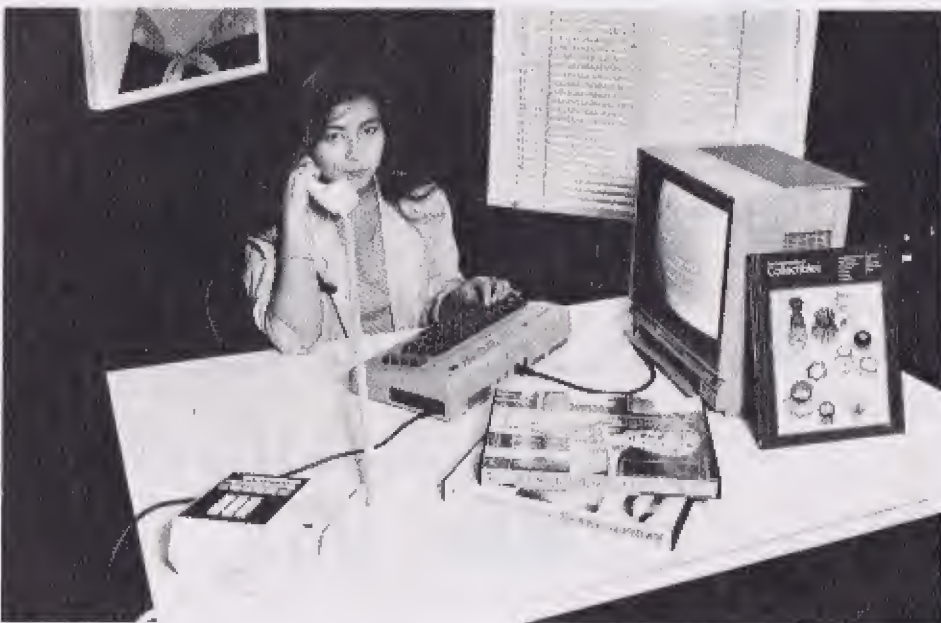
アニメでは、絵を少しずつ動かしながら、フィルムに撮っていくんだけど、パソコンなら、プログラムで絵が動かせそうだなって思ったのネ。

フィルムなら、1秒分作るだけで4から9コマ位絵を描かなきゃいけないから、それは、それは、大変なことなの。

それに比べたら、プログラムの勉強なんか楽なものよ。実際に動かしてみれば、正しいか間違いか、すぐわかるし、直しもキーボードから、ポンポンだからね。フィルムのときなんか、撮影を失敗すると、また最初からやり直したり。この違い、わかってくれる?」

いま、グラフィック・ツールという絵を描くプログラムを作成中とかで、ちょっとデモンストレーションしてくれたけど、これがジツにすばらしい。

キーボードからの命令ひとつで、円を描いたり、線を引いたり、色をぬったり、おまけにその手順から、プログラムが自動的に作られる。Rを押すと、なんと、再現してくれた。テクノ・ギャルのやることは恐ろしいゾ!!

コモドール64¥99,800

大塚美和クン(玉川大芸術学科3年)は絵筆を持ったことはない。でも現在も未来も、自分はアーティストだと思っている。

パソコン・リクルート作戦は、自作のプログラムで、会社分析。

「会社訪問は秋にならないと解禁にならないけど、それから就職活動してたら、まず落ちこぼれ確実ネ」なんて、大学の就職課の先生みたいなことを言うのは、酒井千鶴クンだ。現在、英文科の4年生。彼女が、また驚くことにパソコンを、自由に操るのだ。

そのキーボードさばきは、もうお見事の一言につきる。まるで腕白ボウヤをあやすように、軽いタッチでササッと触れる。

それもそのはず。聞いてみたら、英文ワープロの腕前は、ちょっとしたプロ並みとか。

英文ワードプロセッサを使いたくて、VIC-1001を買ったのが、パソコンとの出会いという。ところが探求心の旺盛な彼女、ワープロだけではもの足りなくなって、ついにBASICに手を出した。なにしろキーボードは英文タイプとキー配置が同じだから、買ったその日から、VICは彼女の右腕となった。

千鶴クンは、現在、自作のリクルートプログラムに、会社のデータをインプットする毎日だ。「小さい頃から、外国で生活するのが夢だったの。だから高校でもESSに入ってたし、その頃から英文タイプもやってた。

今年は、就職の年でしょ。目標は、外資系の会社のセクレタリー。だって、もしかしたら、外国で仕事できるかもしれないじゃない」と千鶴クンは、おっしゃる。

そうなれば、青い目のボーイ・フレンド……?

「いまは、パソコンが面白くてたまらないから、先のことなんて、わからないわ。だって、打てば響くって感じのパソコン、私にピッタリなんだもん。

そういえば、リクルートの中間報告が出たんだけど、コモドールなんてピッタリなの。あなた誰か知ってるんでしょ。紹介してよ」だって。

なかなか積極的なギャルでした。

VIC・1001¥49,800

鈴木央子クン(青学短大1年)はゲーム・フリークだが、いまやプログラミングに挑戦中

酒井千鶴クンは、アノ白百合大学英文科。キャリア・ガールへの道まっしぐら。

VIC! Challenge

遊びのレパートリーがまたふえちゃった!
ごっ機嫌、MAX。

アーチスト気分で、絵を描いてみた
by MAX MACHINE

PROGRAM FOR MAX MACHINE SPRITE CREATE PROGRAM With 2K BASIC

MAX MACHINEは大変多くの機能を持っているため上手く使うととても楽しいゲーム等が作れます。その中の1つがSPRITEです。これを使うと、とても簡単に絵を動かすことができます。

しかしこのスプライトは、形をつくるのに、一枚につき63のデータが必要となり、その計算がとてもめんどうです。そこで画面上でレイアウトをしてデータを出力してくれるプログラムを作ってみました。

〈使い方〉

このプログラムは、MAX·ベーシック·カートリッジが必要です。これはマルチカラー、ノーマルの両方で使えます。

RUNさせると、マルチカラーかどうかを聞いてきます。マルチカラーなら色をきいてきますので、カラーコード(数字)をいれます。

次にレイアウト画面がでてきます。さらにマルチカラーなら、使える色が右上に表示されます。

レイアウト画面は24×21のサークルで示され、左上にカーソルがでてきます。このカーソルをカーソル移動キーで上下左右に自由に動かして、アスタリスク"*"を書き込んでいってください。アスタリスク以外は全てOFF、アスタリスクはONです。マルチカラーでは、CTRL、コモドールシフト十数字で色をかえて、アスタリスクを書いてください。最初に指定した色以外はうけとりませんし、アスタリスク以外もうけつけません。

書き終わったらRETURNを押すと右の2つにノーマルと拡張の2つのスプライトを表示します。さらに"OK(Y/N)?"ときいてきます。画面を修正したければ"N"、これでよければ"Y"を押してください。

"Y"を押すと、"START LINE ?" ときいてきます。このプログラムは、データをリストの形(データ文)で保存するので、そのリストの最初の数字をいれてください。データリストを表示し、プログラムはNEWされますから、そのままRETURNを押しつづけると、リストができます。

あとはTAPEにSAVEすればいいわけです。

〈改良〉

i)多くのレイアウトを一度にしたいならば、720行のNEWをENDにし、START LINEを720より大きくすれば、プログラムはNEWされず、またデータリストと重なりません。

ii)C-64用にするには、230行の6と、590行の6を33にするだけです。

〈プログラムの構造〉

かなり風変わりな方法でデータを作っています。これはTAPEへのシーケンシャルファイルを使ったり、GET文でのカーソルの移動によってプログラムがふくれあがってしまうのをさけたためです。方法を知りたい人は自分で解読してください。

●このプログラムは、あなたが個人として利用する他は、著作権法上、制作者に無断では使用できません。

COPYRIGHT © 1983 H. TANAKA

```
10 REM  ***************************
20 REM  * SPRITE CREATE PROGRAM *
30 REM  *     FOR MAX MACHINE     *
50 REM  *                         *
60 REM  *      (C) 1983,2,20      *
70 REM  *       BY H.TANAKA       *
80 REM  ***************************
90 :
100 POKE53281,1:POKE53280,3:POKE650,128
110 VRAM=1146:VIC=13*4096:CRAM=55419:DIMA(63):A(63)=0
120 COL$="■■■■■■■■"
130 PRINT"■"TAB(12)"■----------------":PRINTTAB(13)"SPRITE CREATER"
140 PRINTTAB(12)"■----------------"
150 IFTTL=1THEN290
160 PRINT"■■■■■■MULTI COLOR OR NOT (M/N) ?"
170 GETMT$:IFMT$="N"THEN:SC=2:GOTO230
180 IFMT$<>"M"THENPRINT"■■■":GOTO160
190 INPUT"■■■■■■■■■■■■SPRITE COLOR ";SC
200 INPUT"■■■■■■■■■■■■■■MULTI 1 ";M0
210 INPUT"■■■■■■■■■■■■■■■MULTI 2 ";M1
220 POKEVIC+28,3:POKEVIC+37,M0:POKEIC+38,M1:REM MULTI COLOR
230 POKE2040,6:POKE2041,6                                  :REM POINTER
240 POKEVIC+23,2:POKEVIC+29,2                              :REM EXPANSION
250 POKEVIC+39,SC:POKEVIC+40,SC                            :REM COLOR
260 POKEVIC+16,3                                           :REM SPRITE MSB
270 POKEVIC,20:POKEVIC+1,99:POKEVIC+2,8:POKEVIC+3,153      :REM PLACE
280 TTL=1:GOTO130
290 FORI=0TO20:PRINTTAB(2)"oooooooo■oooooooo■oooooooo■o"
300 IFINT(I/7)=I/7THENPRINTTAB(2)"■■oooooooooooooooooooooooo"
310 NEXT
320 IFMT$="N"THEN350
330 PRINT"■"TAB(29)"■■■■"LEFT$(COL$,SC+1)"SC■"LEFT$(COL$,M0+1)"M0■";
340 PRINTLEFT$(COL$,M1+1)"M1"
350 PRINT"■"TAB(30)"■■■■┌────┐"
360 FORI=0TO2:PRINTTAB(30)"|    |":NEXT
370 PRINTTAB(30)"└────┘":PRINTTAB(30)"NORMAL"
380 PRINTTAB(29)"■┌──────┐"
390 FORI=0TO4:PRINTTAB(29)"|      |":NEXT
400 PRINTTAB(29)"└──────┘"
410 PRINTTAB(29)"■EXPANDED"
420 INPUT"■■■■■";A$
430 REM DATA MAKING
440 IFMT$="M"THEN500
450 FORI=0TO20
460 FORJ=0TO2
470 FORK=0TO7
480 A=PEEK(VRAM+40*I+J*8+K):IFA=42THENA(I*3+J)=A(I*3+J)+2↑(7-K)
490 NEXT:NEXT:NEXT:GOTO590
500 FORI=0TO20
510 FORJ=0TO2
520 FORK=0TO6STEP2
530 A=PEEK(VRAM+40*I+J*8+K):IFA=42THEN580
540 A=PEEK(CRAM+40*I+J*8+K)AND15
550 IFA=SCTHENA(I*3+J)=A(I*3+J)+2↑(7-K)
560 IFA=M0THENA(I*3+J)=A(I*3+J)+2↑(6-K)
570 IFA=M1THENA(I*3+J)=A(I*3+J)+2↑(7-K)+2↑(6-K)
```

```
580 NEXT:NEXT:NEXT
590 FORI=0TO63:POKE6*64+I,A(I):NEXT                    :REM SPRITE DATA
600 POKEVIC+21,3                                       :REM ENABLE
610 PRINT"■■■■■■■■■■■■■■■■■■■■■■■"TAB(28)"OK (Y/N) ?"
620 GETA$:IFA$="Y"THEN650
630 IFA$<>"N"THEN620
640 PRINTTAB(28)"■            ":FORI=0TO63:A(I)=0:NEXT:GOTO420
650 REM DATA LIST
660 POKEVIC+21,0:INPUT"■■■■■■START LINE ";A:PRINT""
670 FORI=0TO3
680 PRINTA+I*10"DATA";
690 FORJ=0TO15
700 PRINTRIGHT$(STR$(A(I*16+J)),3)",";
710 NEXT:PRINT"■ ":NEXT
720 PRINT"■■":NEW
```

コッポラ監督にも見せたいな「地獄の黙示録」
ミニ・ベーシック・カートリッジでもやれるよ!!

by MAX MACHINE

■レジスターセット及びプログラム1

```
V=13×4096:S=13×4096+4×256
POKE V+21,191
POKE V+28,255
POKEV+ 38,14
POKE V+37,8
POKE V+23,255:POKE V+29,255
POKE V+46,3
POKE S+4,129
POKE S+23,255
POKE S+21,4:POKE S+22,89
POKE S+1,28:POKE S,69
FOR A=0 TO7:POKE 2040+A,6:NEXT
FOR A=2 TO8:POKE V+39+(A-2),4:NEXT
POKE V+1,50:POKE V+3,(78)/80:POKE V+5,110:POKE V+7,(134)/140
POKE V+9,(162)/170:POKE V+11,(190)/200
POKE V+33,0:POKE V+32.0
```

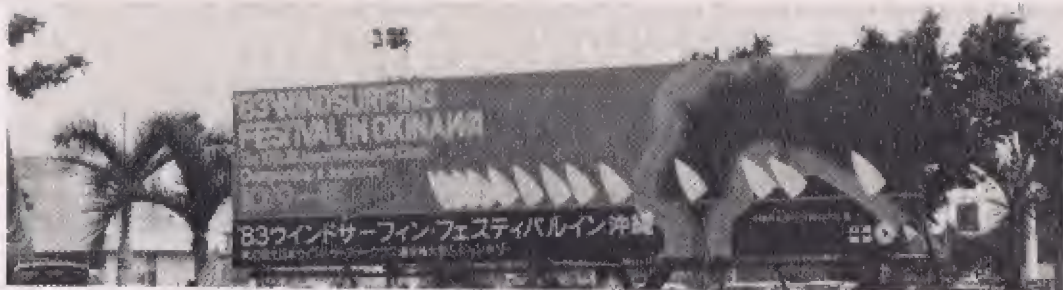
○にするとむずかしくなる。

```
10 FORA  OTO63:READQ:POKE6×64+A, Q:NEXT
20 DATA    0,   0,   0,  85,  85,  64,   0, 128
           0,  10, 168,   0,   2, 160,   0,  62
         168,   0
30 DATA  254, 170, 130, 254, 170, 162, 254, 170
         170, 253,  85,  85, 250, 170, 128,  58
40 DATA  168,   0,   4,  16,   0,  68,  16,   0
          85,  85,   0,   0,   0,   0,   0,   0
50 DATA    0,   0,   0,   0,   0,   0,   0,   0          RUN
           0,   0,   0,   0,   0,   0,   0,   0,   0      CLR:NEW
```

■メインプログラム

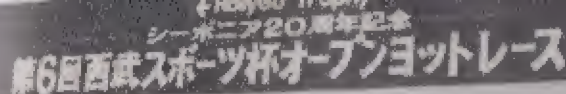
```
  5    PRINT'C'
 10 V=4096×13:S=4096×13+4×256
 20 N=200:M=50:POKE2047, 6: C=0
 30 PRINT
 40 FOR A=1TO255  STEP1.5  :C=C+1
 50 POKE V, A:POKE V+2, 255−A:POKE V+4, A
    POKE V+6, 255−A:POKE V+8, A:POKE V+10, 255−A
 60 B=PEEK(56321):POKE V+30, 0
 70 IF(B AND1)=0 THEN  M=M−1:IF M<49 THEN=49
 80 IF(B AND2)=0 THEN  M=M+1:IF M>220THEN=220
 90 IF(B AND4)=0 THEN  N=N−1:IF N<25 THEN=25
100 IF(B AND8)=0 THEN  N=N+1:IF N>230THEN=230
110 POKE V+14, N:POK V+15, M
120 IF(PEEK(V+30)AND128)>O THEN 400
130 POKE S+4, 128    :POKE  S+4, 129    :POKE S+24, 31
140 NEXT:GOTO 40
400 POKE 2047, 7    :PRINT C:
135 POKE6×64+61, 0:POKE6×64+62, 0
```

MAXマシーン「魔天」ヒュ〜ウゥ〜
ミニ・ベーシック「七番目のゆううつ」のイメージ
ミニ・ベーシック・カートリッジを使用して
チャレンジしてみよう!

```
10  T=54272:REM ==MAX MUSIC SYSTEM=====
    ======BY MAGICAL POWER MAKO========
20  FOR A=0 TO 64988 STEP 1000:B=64988-A
30  H=A/256:L=A-256*INT(H):
    O=B/256:P=B-256*INT(O)
40  POKET+24,15:       POKET+4,1*16+1:
    POKET+11,2*16+1:  POKET+18,8*16+1
50  POKET+5,7:POKET+12,7:POKET+19,7
60  POKET,L:    POKET+1,H:  POKET+7,P:
    POKET+8,O:POKET+14,L:POKET+15,H
70  POKET+4,1*16+0:          POKET+11,2*16+0:
    POKET+18,8*16+0:NEXT:GOTO 20
```

COMMODORE-64も出場した “第10回ウインドサーフィン全日本選手権大会” in OKINAWA。

もの思う秋、なのにこの秋のオキナワは、夏にも増して騒がしくなった⁉
そのわけ①10月15日から19日まで、あの世界選手権大会が開かれたのと同じ沖縄海中公園で、“第10回ウィンドサーフィン全日本選手権大会”が行なわれたんだ。
そのわけ②今大会の得点集計とランキングの作成に、COMMODORE64がオフィシャルコンピュータとして出場したのです。
それで、7日間に渡っての「熱き男の闘い」を見守る目、喝采の声、拍手…と、熱気ビシビシのオキナワでありました。
ところで、コモドールのコンピュータと世界選手権とは、'81年以来のおつき合いが続いていて、ますます仲むつまじくなるもよう。これからの二人三脚的活躍、大いに期待しちゃうよね！

MFU/日本メンズファッション協会のショウに、MAXが大活躍！

去る7月15日、「感動新世紀― SENSATIONAL NEW ERA」と銘うった秋・冬メンズファッションショウが開催されました。
豊かになった暮しの中で、物にふり回されるバカバカしさに気づき、意識の革命までもともなったファッション界の素材イノベーション。その精神はコンピュータにも通じるもの。というわけで、今回MFU(日本メンズファッション協会)とコモドールのタイアップとなったわけです。
コシノジュンコから、POLO、ダーバン、そしてデサントまで。華麗に繰りひろげられる男の着こなし、男の風景(シーン)。
ツィード、コーデュロイ、ニット…素材そのものを見つめ直す提案が、ショウ全体に満ち、観る人をひきこみました。
会場入口には、唯一正式に認定されたコモドールのMAXがおめみえ。
そして、「アンケートにご協力いただきました方の中から、コモドールジャパンのご協賛により今話題の“マックス・マシーン”をソフトウエア付きで差し上げます」の、お楽しみつき。
アナウンスが流れたとたん、会場から「わっ、ホシイ！」の声がもれました。
コンピュータの素材、たとえばトランジスタ、IC、LSI…その1つ1つをおろそかにせず、特性を最大限にひき出す努力と研究。これが、コモドールが世界を舞台に駆け回る大きな原動力となっているのですね。

西武百貨店主催
第6回 西武スポーツ杯 オープンヨットレース with COMMODORE-64

5月22日(日)、まさにヨット日和り。太陽がニッコリほほえんだこの日、第6回西武スポーツ杯オープンヨットレースが和田・長浜海岸で開催されました。

すでに三浦半島のヨットマンたちには定着しているこの大会。55年・秋を皮切りに、毎年春・秋に催され、回を重ねて第6回目。参加艇の数も今大会は261艇と、この春のヨットレースでは日本一を記録したとか。

加えて、大会本部のシーボニア・マリーナが開業20周年記念ということもあって、21日(土)夜から盛大な前夜祭も行なわれて、もう、盛りあがりっぱなし。

さて、レースは10時スタートの第1レースと12時スタートの第2レースに分かれ、クラス別参加艇はすべてのディンキー対象という大会主旨のオープンヨットレースであり、オリンピックコースシステムを採用した本格的なもの。

今回はコモドールコンピュータのCOMMODORE-64が正式に採用され、前日一日がかりで参加申告艇のNAME, ENTRY NUMBER, RACE NUMBER, FLEET, CLASSなどの打ちこみにオフシャル・コンピュータとして大忙し。

さらにレースが進行すると、続々ゴールインする各艇の得点、ランキングの集計にもCOMMODORE-64が本領を発揮してスピーディに処理していきます。

PM4:00からは表彰式および閉会式。今大会の話題は毎回出場の柳さん親子、小学生の息子さんが、見事ミニホッパークラスで優賞したこと。柳さんもニコニコご満悦そうでした。

こうして、ひときわにぎわった海岸に陽が落ち、各クルーも大会関係者も快い疲労感に酔いながら、今大会の幕はおりたのでした。

VIC! TOPICS

朝日パーソナルコンピューターショー'83
ビギナーも上級者も、大人も子供も、コモドールコンピュータに全員が胸ワクワクだった4日間。

「朝日パーソナルコンピューターショー'83」が9月15～18日の4日間、池袋サンシャイン文化会館で開催されました。サブタイトルは、今年の世界コミュニケーション年にちなみ"パーソナルコミュニケーション新時代"。
総勢80,000人もがドッと繰りだしたこのショーで、いちやく注目を集めたのは3Fミプロ展示場の㈱ムーミンのブース。
"パソコンはゲームを楽しむためだけのものではなく、ビジネス、遊び、実用、夢…と使いこなしてこそ未来へつながる"の主旨で、VIC-1001, MAX MACHINE, COMMODORE 64, そして新製品のSX64(参考商品)とコモドールコンピュータが勢揃い。そして、それぞれのもち味を生かして、VICでNゲージのコントロールや電話回線を使っての大阪との交信、C-64ではパソコンを知るための「INTRODUCTION TO BASIC」やビジネスでの新分野、家族で楽しめるゲーム、などなど。さらにビデオ、モニター、外国雑誌などのオモシロ情報コーナーと、ビギナーから上級者まですべてに満足のいく設定は、さすが。
さて、反応はいかに?と見ると、「情報を音に変えて受話器に送るテレフォンカプラーの仕組みを教えて!」とか、「LOGOってやさしいのですか?」とか係への質問が飛び交っていました。
こうして関係者一同、汗だくの応対の甲斐あって、大盛況のうちに4日間の幕を閉じました。

ニットファッションの分野でもCOMMODORE 64がキビキビと活躍中！

「うちは、ニューヨークや南米ウルグアイ、イタリー、西ドイツ、アイルランドからニット製品そのものまたは糸を輸入して製品にし国内の専門店に収めたり、また海外に輸出したりする貿易商社。商品の売れ筋やどの品番がどれだけ売れたか、在庫はどれだけか、などの在庫管理にコンピュータは必携。COMMODODORE64はもう僕の片腕ですよ。」
アジア・アメリカン貿易㈱国内マーケティング部の倉田さんはこう語ってくれた。
「ここのニットって、色調がとても微妙で同じ柄でも少しずつ違うの。糸をつむぐ、染める、編むのすべてハンドメイドという手作りの味かしら。とにかく私のお気に入りです。それにメンズものもあるので、彼とペアで楽しめて、大感激。」とは、ファンのお嬢さんの声。この冬のおしゃれは、ニットでビシッと決めたいネ。COMMODORE64も影ながらお手伝いしちゃうヨ！

さ〜すがアメリカだネ。
コモドールのコンピュータはひかり輝く我らのコルゲートハミガキ!?
というわけで米国誌の紹介です。

COMMODORE A

VIC-1001シリーズ

COMPONENT SYSTEM

MODEL NO.	DESCRIPTION	PRICE(¥)
VIC-1001	カラー・パーソナル・コンピュータ	49,800
VIC-1010	エクスパンション・モジュール	29,800
VIC-1110	8K RAM・カートリッジ	9,800
VIC-1111	16K RAM・カートリッジ	14,800
VIC-1211	スーパー・エクスパンダー・カートリッジ	9,800
VIC-1211M	スーパー・エクスパンダー(3KRAM付)・カートリッジ	14,800
VIC-1212	プログラマーズ・エイド・カートリッジ	9,800
VIC-1213	マシン・ランゲージ・モニター・カートリッジ	9,800
VIC-1011A	RS-232C・アダプター・カートリッジ (ターミナル・タイプ)	9,800
VIC-1011B	RS-232C・アダプター・カートリッジ (カレント・ループ・タイプ)	9,800
VIC-1801	ベーシック・フォア・ザ・VIC (VIC用BASIC学習プログラム カセットテープ	3,800
1701	カラーモニター(モニター・ケーブル付)	72,800
1525	グラフィック・プリンタ(80桁)(接続ケーブル付)	69,800
プリンタ・リボン	1525プリンタ用リボン	2,500
C2N	カセット・ドライブ	14,800
1541	シングル・インテリジェント・フロッピー・ディスク(接続ケーブル付)	79,800
1311	ジョイ・スティック	3,800
1312	パドル	4,800

GAME SERIES

VIC-1910	レーダーラットレース	2,800
VIC-1911	スカイフォール	2,800
VIC-1912	モグラタタキ	2,800
VIC-1913	レイダース	2,800
VIC-1919	チェス	2,800
VIC-1921	スーパースマッシュ	2,800
VIC-1923	ゴーフ	2,800
VIC-1924	オメガレース	2,800
VIC-1925	マネーウォーズ	2,800
VIC-1931	クラウンズ	2,800
VIC-1932	ガーデンウォーズ	2,800
VIC-1933	ビンゴ	2,800
VIC-1937	シーウルフ	2,800

●製品改良のため価格・仕様は予告なく変更することがあります。 (1983年3月15日現在)

LL LINE PRICE LIST

COMMODORE64シリーズ

COMPONENT SYSTEM

MODEL NO.	DESCRIPTION	PRICE(¥)
COMMODORE 64	カラー・パーソナル・コンピュータ	99,800
C2N	カセット・ドライブ	14,800
1525-J	プリンタ（接続ケーブル付）	69,800
1541	フロッピー（接続ケーブル付）	79,800

SYNTHESIZER CARTRIDGE SERIES

MODEL NO.	DESCRIPTION	PRICE(¥)
3201	ミュージック・コンポーザー	3,800
3202	ミュージック・マシーン	3,800（4月末発売）

GAME CARTRIDGE SERIES

MODEL NO.	DESCRIPTION	PRICE(¥)
3401	オメガレース	2,800
3402	ウイザード・オブ・ウォー	2,800
3403	キックマン	2,800
3502	ジュピターランダー	2,800
3504	レーダーラットレース	2,800
3505	ロードレース	2,800
3508	モールアタック	2,800

●製品改良のため価格・仕様は予告なく変更することがあります　（1983年3月31日現在）

MAXマシーンシリーズ

商　品　名	単　価（¥）
マックス・マシーン本体	34,800
ゲーム・ミニベーシックシリーズ（カートリッジ）	
オメガ・レース	2,800
アヴェンジャー・	2,800
ジュピター・ランダー	2,800
レーダーラット・レース	2,800
ロード・レース	2,800
モール・アタック	2,800
スーパー・エイリアン	2,800
クラウンズ	2,800
マネー・ウォーズ	2,800
ウイザード・オブ・ウォー	2,800
キックマン	2,800
コンピュータベーシックシリーズ	
ミニベーシックⅠ	3,800
マックスベーシック	未定
シンセサイザーシリーズ	
ミュージックコンポーザー	3,800
ミュージックマシーン	3,800

●製品改良のため価格・仕様は予告なく変更することがあります。

COMMODORE 64

VIC-1001 COMPONENT SYSTEM

フロンティア精神が生んだ革新の技術、完璧な

VIC-1001はコンポーネント・システムを

VIC-1010

エクスパンション・モジュール

各種オプションを接続するためのシステム拡張用ボード。VIC-1001のメモリー・エクスパンション・バスにダイレクト接続。22ピンのエッジカード・コネクター6本実装。VIC-1110、1111、1211(1211M)、1213を複数使用の場合に使用。最大6個までのパックを同時に装着することができます。

VIC-1011A

RS232Cアダプター・カートリッジ
(ターミナル・タイプ)

ユーザー・ポートに接続することにより、RS232Cターミナル・タイプ・インターフェイスをもった各種機器が接続可能になります。音響カプラを接続すると、電話線を使って、遠く離れたコンピューターどうしのコミュニケーションができます。

VIC-1011B

RS232Cアダプター・カートリッジ
(カレント・ループ・タイプ)

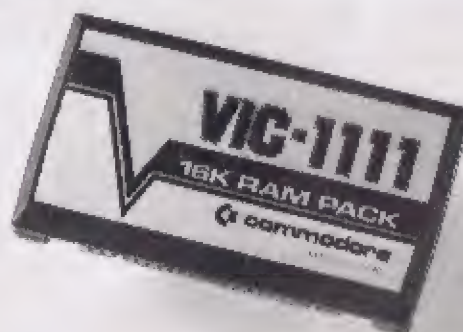
ユーザー・ポートに接続することにより、RS232Cカレント・ループ・タイプ・インターフェイスをもった各種機器(テレタイプなど)が接続可能になります。

VIC-1110

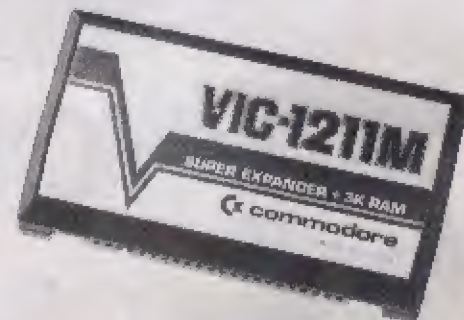
8KRAMカートリッジ

拡張用メモリー・パック。8Kバイトが実装されており、他の拡張メモリーを併用しない場合、VIC-1001のユーザー・エリア(3583バイト)と合わせて、11775バイトのユーザー・エリアとなります。メモリー・エクスパンション・バスに直接接続するか、VIC-1010エクスパンション・モジュールのコネクターに接続。

VIC-1111

16KRAMカートリッジ

拡張用メモリーパック。16Kバイトが実装されており、他の拡張メモリーを併用しない場合、VIC-1001のユーザー・エリア(3583バイト)とあわせて、19967バイトのユーザー・エリアとなります。メモリー・エクスパンション・バスに直接接続するか、VIC-1010エクスパンション・モジュールのコネクターに接続。

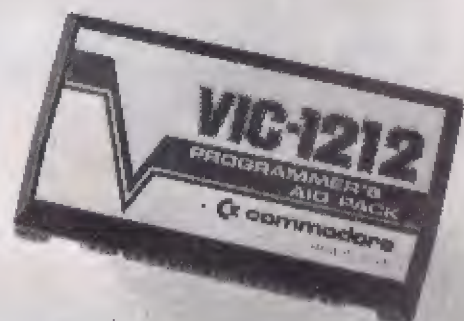
VIC-1211M

スーパー・エクスパンダー 3KRAM付きカートリッジ

ハイレゾリューション(高解像)グラフィック・コマンド、サウンド・コマンドその他をもつBASIC拡張パック(3KRAM付き)。エクスパンション・バスまたはVIC-1010に接続。ハイレゾ・コマンドをリリースすれば、そのまま3KRAMパックとして使用可。

3KRAMなしのVIC-1211スーパー・エクスパンダーは、VIC-1010に接続し、増設RAMパックを併用することが必要。

追加される BASIC コマンド。HIRES, PLOT, BOX, CIRCLE, SETC, SOUND, MUSIC, TEMPO, PIANO, LOCATE, CHAR, PDL, LIGHT, JOY, RELEACE など。ファンクションキーに12種のコマンドをアサイン(ユーザーが変更することも可能)。コントロール・ファンクションを追加。

VIC-1212

プログラマーズ・エイド・カートリッジ

BASICのプログラミング、プログラムのデバックを大幅に効率化する"TOOLKIT"のコモドール版。プログラミングおよびデバック(エディト)用の各種BASICコマンドを追加。RENUMBER, MERGE, FIND, CHANGE, DELETE, AUTO, TRACE, STEP, DUMP,

M LINE UP

周辺機器

主張します。

HELPなど。ファンクションキーに2モード計24種のコマンドをアサインしています（ユーザーが変更することも可能）。コントロール・ファンクションを追加。エクスパンション・バスまたはVIC-1010に接続。

VIC-1213

マシン・ランゲージ・モニターカートリッジ

VIC-1001での機械語プログラムの作成、デバックを大幅に効率化するツールキットです。①キーボードからマシン・ランゲージ・コマンドを入力することによって、CPUの動きをコントロールでき、システムをファイルできます。②CPUのステップ動作が可能です。③レジスタ、プログラムステータス、スタックポインタおよびメモリの内容を自由に変更できます。

1701

専用カラー・モニター(モニター・ケーブル付)

コンポジット・ビデオ・インプット。スピーカー内蔵。14インチ。モニター・ケーブルで、VIC-1001のカラー・ビデオ・インターフェイスに接続。

1525

グラフィック・プリンタ

ドット・マトリックス・インパクト・プリンタ。80桁。印字スピード30字／秒。VIC-1001のもつすべての文字(英数字、グラフィック文字、カタカナ)を印字できます。標準文字印字、横2倍文字印字、プログラマブル・グラフィック印字が可能。VIC-1001シリアル・バスまたはVIC-1540シリアル・バスに付属ケーブルで接続。

C2N

カセット・ドライブ

記憶容量、約160Kバイト(C-30使用時)。記憶方式はコモドール方式（500ボー）。VIC-1001のカセット・インターフェイスに接続。SAVE中のランプ装備付。

1541

シングル・フロッピー・ディスク・ドライブ

インテリジェント・フロッピー・デスクで、VIC-1001のメモリーを使いません。小型ながら、170Kバイトものデータを高速処理。大容量ファイルとして使用でき、カセットベースに比べて飛躍的にデータ処理ができるようになります。使用ディスケット：5.25インチ標準ミニ・フロッピー(ソフト・セクター)。記憶容量175Kバイト／ディスケット。ディレクトリー・エントリー：144／ディスケット。セクター／トラック：17~21。バイト／セクター：256。トラック：35。ブロック：683。
VIC-1001シリアル・バスに付属ケーブルでダイレクトに接続。(接続ケーブル付)

1311/1312

ジョイスティック／パドル

ゲームをいっそうゲームらしくできるツール。

VIC GAME SERIES

VIC-1910	レーダー ラットレース
VIC-1911	スカイ フォール
VIC-1912	モグラ タタキ
VIC-1913	レイダース
VIC-1919	チェス
VIC-1921	スーパースマッシュ
VIC-1923	ゴーフ
VIC-1924	オメガ レース
VIC-1925	マネーウォーズ
VIC-1931	クラウンズ
VIC-1932	ガーデン・ウォーズ
VIC-1933	ビンゴ
VIC-1937	シーウルフ

新種ゲーム続々登場!!

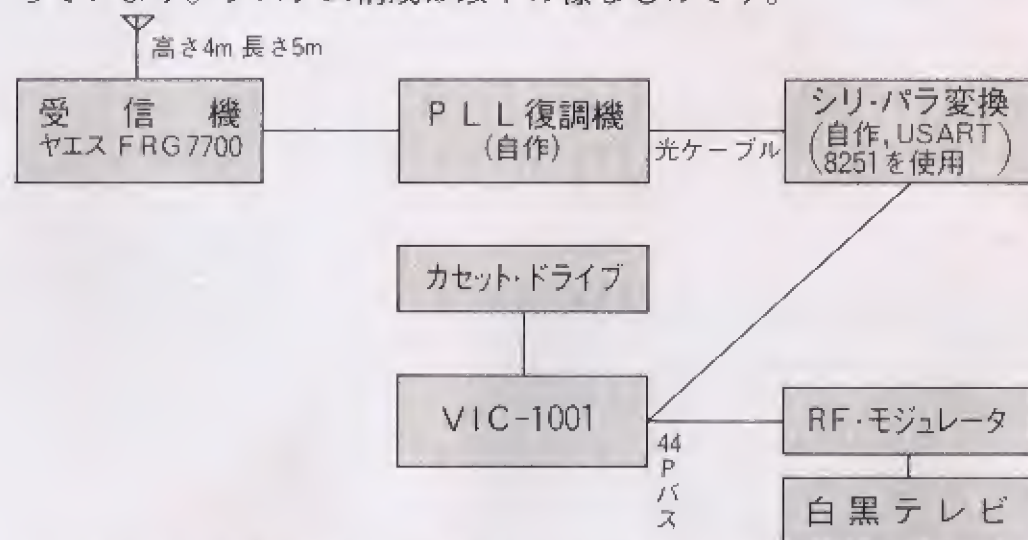
V I C ! M E M

PERSONAL COMPUTER VIC DIET!

●じっくりと根気を持ってチャレンジしよう
●無拡張システムからまずはじめてみましょう。

最近のパーソナルコンピュータシステムは目に余る程の拡張BASICが搭載されているようです。コモドール社が、ワンボードコンピュータ「K2M-1」を発売して十余年。
初期のKIM-1が、機械語しか使えないワンボードコンピュータだったのに対して、今のパーソナルコンピュータは、BASICは使える、色は16色出る、シンセサイザ級の音は出る、等々いろいろな機能がついているのは、ものすごい進歩であると思います。しかし、どれだけのユーザーが、自分の持っている、コンピュータの機能を使いこなしているのだろうか？たぶん数える程しかいないのでは？昔のユーザーはマニアであったが、今は必ずしもそうだとは限らない。時代の流れと言ってしまえばそれまでかもしれない。
それでは、何故ユーザーが使いこなしていないのだろう？それは先に述べた、目に余る程の拡張BASICではないか？今のパソコンは、「でもできます、でも何もできません」の一言で言い表せる程中途半端なのである。そして、拡張BASICというのは命令の数をいたずらに増やしているだけではないのだろうか？命令が増えれば、それに応じて覚える量が増える、そして命令形態も覚えなければならない。そのうち覚えきれなくなり、パソコンへの興味もうすれていく。これらの拡張BASICの大部分は、POKE命令等で処理出来るものである。また、多機能というのも、要因の一つではないかと思う。音、色、ハイリゾ画面、プリンタ・ディスク等々一度に覚えられるものではないと思う。音なら音だけ、色なら色だけというのならば、確実に覚えることが出来るでしょう。そこで、私は「無拡張システム」をおすすめします。「VIC-1001」ならば、本体だけ「MAX-MACHINE」ならば、MAX-BASICカートリッヂだけ、「C-64」ならば本体だけという最小のBASICシステムから何かを学ぼうということです。丁度良いことに、コモドール社のパーソナルコンピュータは、カートリッジを差し込まなければ、標準BASICしか動かないのです。そして、段階的に、パーソナルコンピュータシステムの機能をマスターして行くのです。まず、本体のみ標準BASICを100%使いこなしましょう。次に自分の一番マスターしたい機能の使いこなしを勉強しましょう。その機能のマスターの仕方も最初からカートリッジの拡張BASICに[illegible]ではなく、標準BASICの命令を使ってその機能を充分に発[illegible]ることを知ってから、カートリッヂをマスターするのです。
そうすれば、貴方もBASICに関してはプログラマーです。BASICはこわいものなしです。
「無拡張システム」というのは、勉強するには、非常に強い教材なのです。この「無拡張システム」で、BASICの勉強をしなおしてみませんか？貴方のパソコンライフに、このことが、ヒントの一つとなれば、幸いです。

VOICE!

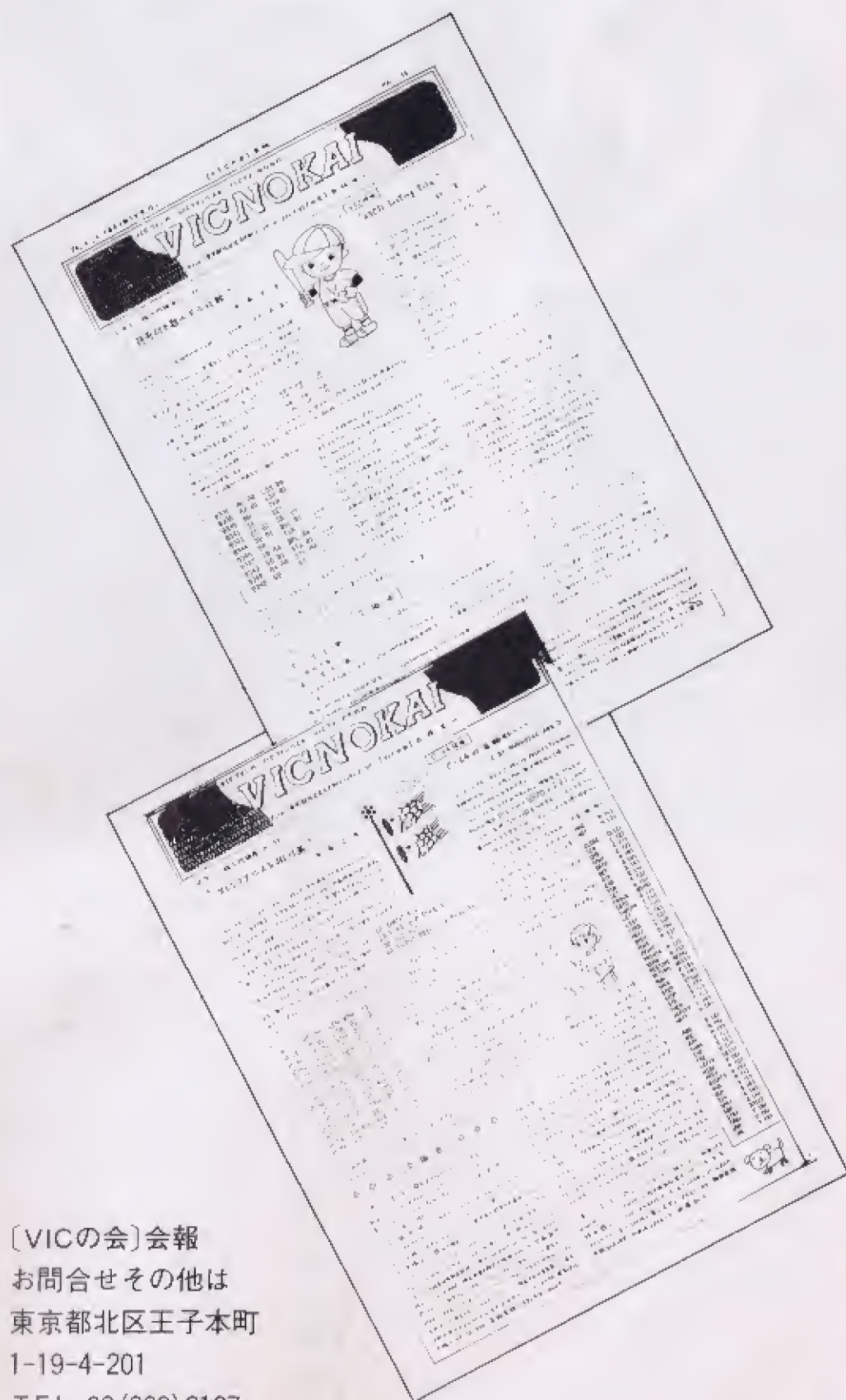
残暑きびしきおりながら、VIC編集部の皆様にはお元気でお過ごしの事と存じます。
私は昭和56年5月に北大生協でVIC-1001を買ったUSERです。
現在VICを主としてラジオ・テレタイプ(テレックス)の解読に使っています。システム構成は以下の様なものです。

これまでに受信できたRTTY局は以下の通りです。
アマチュア局多数、気象通報局多数、時事、共同、新華社、UPI、TASS、AFP、CNA（台北）、KCNA（平城）、YONHAP（京城）、ANSA（シンガポール）、PL（ハバナ）、ノーボスチ通信、INA（バグダット）ADN（ベルリン）、KUNA（クウェート）、CETEKA（プラハ）、VNA（ベトナム）、IINA（ローマ）、DPA（ドイツ）、その他内容のよくわからない局等です。('83.7.26～8.11の調査)。VICから受信機に入るノイズを減らすためTV受像器にはハイパス・フィルターを付けラインフィルターをVICに付け、オプティカル・ケーブルを使って受信機とVICを電気的に切り離したり、いろいろ苦労していますが、まだかなりのノイズが有ります。良い対策が有ったら御教示下さい。

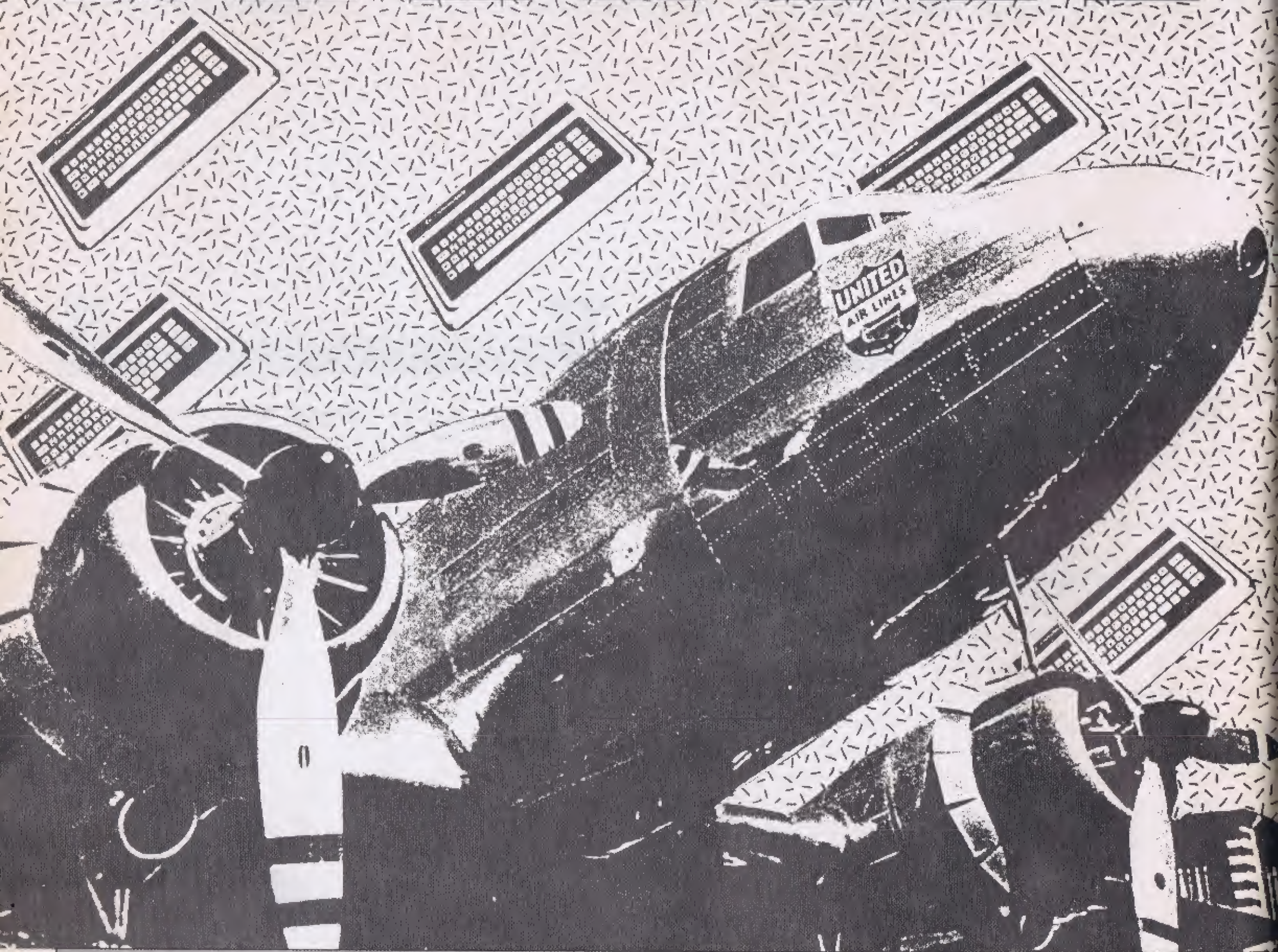
〒063 札幌市西区発寒8条7丁目563番地
樋口 琢也(24才 tel 011-661-8702)

COMMODORE MESSAGE

●HAが創り出すクリエイティブ・ライフ!!
かって、現代ほどのスピードで生活革命が、進行したことが、あったでしょうか。軽・薄・短・小と言われるモノの簡便化に伴って、私たちのくらしは猛烈な勢いで、質変化しようとしています。
科学の進歩が、生活の断面の一つひとつに大きな影響を与え、私たちは次から次へと様々な雑務から解放され始めています。
ビジネス面での人の解放は、その最も顕著な例の一つに挙げることが、できます。OA化は、賛否両論、種々様々な意見の渦巻く中で、着実に進行しているのです。ある人は、その浸透のスピードの前でたじろぎ、うろたえ、ある人は事態の変化を認め、それに追いつかんがために多大な努力を払っているかもしれません。
また、人によってはある種の強迫観念を抱いている人もあるでしょう。しかしそれもこれも、対処すべき相手が、かつてあまり接したことがなく、妙に気むずかしい印象を与えるコンピュータというシロモノだからだと言えないでしょうか?
私たちコモドールは、むしろ生活のヒダの一つ一つに既に入り込んでいるコンピュータの存在を認め、そこに生じるメリットを積極的に評価する所からくらしをもう一度見つめてみたい、そう考えています。あくまでも"人のくらし"を見つめ、そこに主眼を置いてコンピュータを考えていくという姿勢。それを大切にしたいと考えているのです。
H. A. (HOME AUTOMATION)が私たちコモドールの企業ポリシーで、あり、企業目標であり、製品開発コンセプトでもあるのは、そこに理由があるのです。徹底した暮しの簡素化。所詮、機械であり、機械だからこそ様々な活用法を見出すことのできる。コンピュータを利用することによって、それを私たちが、手に入れた時、実は、私たちの暮しは確実に前進するはずなのです。
なぜなら、人の欲求の中で最も高度な欲求とされる自己実現欲求を人それぞれのクリエイティブな発想で、現実のものにしていくことができるだけのゆとりを私たちは獲得できるからです。
フリーにそしてクリエイティブに使える時間。人は常にそれを求めつづけてきたと言っても過言ではないでしょう。
くらしを豊かにする、ということはくらしを簡素化し、しかも便利にし人の関心と努力をできるだけクリエイティブな方向に向かわしめることだ。と私たちコモドールは確信しているのです。H. A。ホームオートメーション!!それは、人の暮しを真の意味で人らしくするための、コモドールなりのアプローチなのです。

VIC COMMUNITY

VICの熱烈ファンから声が届きました!

〔VICの会〕会報
お問合せその他は
東京都北区王子本町
1-19-4-201
TEL. 03(909)6197
白根栄一さんまでどうぞ。

販売店LIST

VIC-1000SERIES取扱店

●札幌——北海道

九十九電気札幌店
065 札幌市中央区南二条西3丁目15-1/ 011-241 2299(代表)

バグハウス札幌
064 札幌市中央区南3条西4丁目エイトビル内／011-261-5451

協栄産業㈱北海道支店
060 札幌市中央区北6条西24-5／011-642-6101

●青森——東北

㈱電技パーツ
030 青森市中央1-21-15／0177-77 4141

●秋田

㈲電子センター秋田
010 秋田市大町6-1-16／0188-64 6058

●宮城

㈱シー・ティ・エス
980 仙台市中央4-8-3(宮城食糧会館2F3号)／0222-66-2061

兼松家電販売仙台営業所
982 仙台市河原町1-3-54

●長野——信越

岡谷バイト・ショップ
394 長野県岡谷市幸町6-11(五十川ビル)／ 02662-3-1075

●新潟

㈱エス・エフ・シー新潟
951 新潟市関屋田町1 13河野ビル1F／0252 66-2233

●群馬——関東

伊勢崎バイト・ショップ
372 伊勢崎市今井町755／0270 23 2302

●栃木

㈱トヨムラ宇都宮店
320 宇都宮市宿郷町365-7／0286-36-5315

協栄産業筑波出張所　305 茨城県新治郡桜村竹園3丁目
竹園ショッピング センター202号／0298-51-1364

●埼玉

西武百貨店　大宮店マイコンコーナー
330 大宮市宮町1-60／0486-42 0111

㈱トヨムラ大宮店
330 大宮市宮原町3 515 2／0486 52-1831

㈱トヨムラ川口店
332 埼玉県川口市芝2-25-3／0482-68-7826

●千葉

西武百貨店船橋店8F マイコンコーナー
273 船橋市本町1 2 1 ／0474 25 0111

船橋そごう3Fマイコンショップ
273 船橋市浜町2 1-1　0474 33 3711　内(2339)

㈱ダイエーホームワールド船橋マイコンコーナー
273 船橋市浜町2-1 1／0474 34-3181

●東京

関東バイトショップ
101 千代田区外神田1-15-16(ラジオ会館4F)／03-253-5264

アキハバラバイトショップKOYO
101 千代田区外神田1 15-16(ラジオ会館7F)／03 255 6504

真光無線㈱
101 千代田区外神田1-15 16(ラジオ会館8F)／03 255 0450

㈱COM　101 千代田区神田佐久間町1 8 4(ニュー千代田ビル)
／03 251 1523

㈱トヨムラ　東ラジ店
101 千代田区外神田1-10-11／03-253-4693

パスカル
101 千代田区外神田1-15-16(ラジオ会館4F)／03-255-4657

上新電機Jap渋谷店
150 渋谷区道玄坂2-28-4／03-496-4141(代表)

松屋㈱銀座本店
104 中央区銀座3-6マイコンコーナー／03-567-1211

㈱トヨムラ新宿店
160 新宿区西新宿2-4-1 NSビル5階／03-342-1571

石丸電気株式会社マイコンセンター
101 東京都千代田区外神田1·9·14／03·255·3111(代表)

九十九電機㈱
101 千代田区外神田3-1-14／03-251-0531

九十九電機ニュー秋葉原センター店
101 千代田区外神田1-16-10／03·251-0986

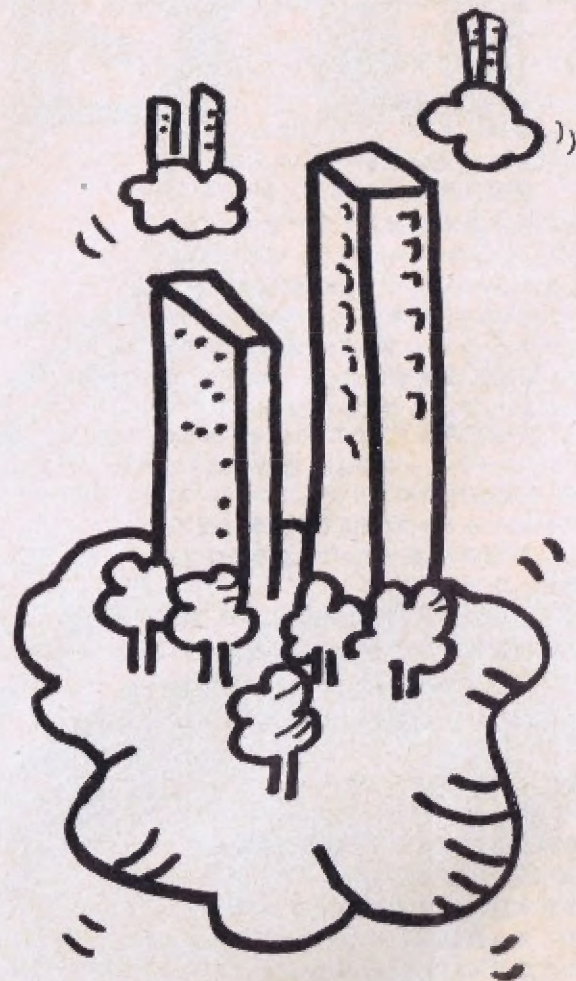
九十九電機7号店
101 千代田区外神田1-9-9／03-253-4199
シーガル㈱
192 八王子市中町7-7(西川ビル3F)／0426-25-9960
サンエイパーツ
185 国分寺市南町3-22-31／0423-23-2441
Tmdシステム巣鴨
170 豊島区巣鴨1-12-6／03-941-8621
西武百貨店池袋店9Fマイコンコーナー
171 豊島区南池袋1-28-1／03-981-0111
西武百貨店渋谷店マイコンショップ
150 渋谷区宇田川町21-1／03-462-0111
マイコン・ショップCSK(新宿住友ビル店)
160-91 新宿区西新宿2-6-1(新宿住友ビル37F)／03-342-5299直通
マイコン・ショップCSK(新宿西口店)
160 新宿区西新宿1-12-18／03-342-1901代表
マイコン・ショップCSK(渋谷店)
150 渋谷区神南1-20-9スタジオパルコ5F／03-477-5875
マイコン・ショップCSK(吉祥寺店)
180 武蔵野市本町1-5-1吉祥寺パルコBF1／0422-21-8111
マイコン・ショップCSK(新宿東口店)
160 新宿区歌舞伎町1-5-4 第6荒井ビル／03-207-3071
キディランド原宿店
150 渋谷区神宮前6-1-9／03-409-3431代表
伊勢丹新宿本店アダルト・ホビー・コーナー
160 新宿区新宿3-14-1／03-352-1111代表
マイコン・ベース銀座－オムロン・マイコン・システムズ
104 中央区銀座1-8-21／03-535-3381
東武池袋本店7F・玩具売場
171 豊島区西池袋1-1／03(981)2211
東急東横店東4F・文具事務用品売場
150 渋谷区渋谷2-24-1／03-477-4431
東急本店3F・文具事務用品売場
150 渋谷区道玄坂2-24-1／03-477-3468
大丸東京本店7F・玩具売場
100 千代田区丸の内1-9-1／03-212-8011
そごう有楽町本店6F・玩具売場
100 千代田区有楽町1-11-1／03-284-6711
東急吉祥寺店6F・文具事務用品売場
180 武蔵野市吉祥寺本町／0422-21-5111
大丸八王子店6F・玩具売場
192 八王子市横山町18-5／0426-26-1111
東急町田店8F・玩具売場
194 町田市原町田6-9-7／0427-28-2384
小田急町田店5F・アダルト・ホビー・コーナー
194 町田市原町田6-12-20／0427-27-1111
伊勢丹浦和店5F・玩具売場
336 浦和市高砂1-15-1／0488-24-5111
高島屋柏店6F・玩具売場
277 柏市末広町3-16／0471-44-1111
伊勢丹吉祥寺店5F・玩具売場
180 武蔵野市吉祥寺本町1-11／0422-21-1111
ミドリヤ渋谷店音と映像フロア
150 渋谷区道玄坂2-29-5／03-463-3411(代)
ダイエー碑文谷店
152 目黒区碑文谷4-4-1／710-1111(代表)
㈱トヨムラTMDシステム
101 千代田区外神田4-4-1／03-253-5754
トム・システム
103 中央区日本橋人形町1-9-2冨士ビル／03-667-2138
●神奈川
㈱工人舎
231 横浜市中区松影町2 7 21／045 662 0688
アイテムコンピュータシステム
251 藤沢市藤沢136 日の出ビル2F202号 0466 27 1668
西武百貨店藤沢店6F・マイコン・ショップ
251 藤沢市本町2-1-17／0466-27-0111
トヨムラ横浜店
232 横浜市中区松影町1-3-7／045-641-7743
横浜高島屋玩具売場(テレビ・ゲーム・コーナー)
220 横浜市西区南幸1-6-31／045-311-1251
●山梨――――――――――中部
中込電機商会
400 甲府市丸ノ内2-4-20／0552-24-5431
●静岡――――――――――東海
ヘルツ電子工業㈱
433 浜松市小豆餅2-29-8／0534-37-5915
㈱トヨムラ静岡店
422 静岡市八幡1-4-36／0542-83-1331
西武百貨店浜松店マイコンショップ
430 浜松市鍛治町15／0534-55-0111
マルツ電波
430 浜松市板屋町390／0534-54-2366
㈱サンワード
430 浜松市砂山町328-11駿河銀行ビル／0534-56-2100
●愛知
カトー無線パーツ㈱
460 名古屋市中区栄3-32-28／052-262-6471
九十九電機名古屋店
460 名古屋市中区大須3-30-86(ラジオセンター名古屋3F)
052 263 1681
名古屋バイトショップ
460 名古屋市中区大須3-30-86(ラジオセンター名古屋3F)
／052-263-1629
㈱トヨムラ名古屋店
460 名古屋市中区大須3 30 86(ラジオセンター名古屋2F)
052 263 1661
●三重
理工産業(四日市)
510 四日市市九の城町4 20 0593 51 1651
理工産業(松坂)
515 松坂市舟江町785／0598-51-4312
●富山――――――――――北陸
無線パーツ高山店
930-11 富山市布瀬町二区759 4／0764 21 6822
無線パーツ高岡店
933 富岡市永楽町2 4／0766 25 6822
●石川
無線パーツ金沢店
921 金沢市西泉町2-28／0762 44 3070
㈱ダイエー金沢店
920 金沢市尾張町2-1-1／0762-23-7111
●福井
マルツ電波
910 福井市豊島2-7-4／0776-21-2360
●大阪府――――――――――近畿
共立電子産業㈱コムスポット共立
556 大阪市浪速区日本橋5-12-19(日本橋会館2F)
／06-644-4666
松下電器貿易㈱
542 大阪市南区南船場4-3-2／06-282-5604
システム応用研究所
545 大阪市阿倍野区阪南町1-45-2／06 624 7829
阪急百貨店マイコンコーナー
530 大阪市北区茶谷町1-27／06-375-1231
高島屋大阪店 マイコン・コーナー
542 大阪市南区難波新地6番町14番地／06-631-1101
㈱マイクロコンピュータシステムズ
531 大阪市大淀区長柄西1 4 12(佐藤ビル2F)
／06 358 3045
大丸百貨店大阪店
542 大阪市南区心斎橋筋1-118／06-271-1231
近鉄百貨店阿倍野店
545 大阪市阿倍野区阿倍野1-1／06-624-1111
上新電機㈱Japメディアランド
556 大阪市浪速区日本橋4-11-3／06-644-1613
上新電機㈱J＆P
556 大阪市浪速区日本橋5-6-7／06-644-1413
上新電機㈱三番街店
530 大阪市北区芝田1-1-3／06-372-6912
上新電機㈱一ばん館
556 大阪市浪速区日本橋5-1-11／06-644-1813
上新電機㈱五ばん館
556 大阪市浪速区日本橋4-12-4／06-644-1513
上新電機㈱3ばん館
556 大阪市浪速区日本橋3-7-5／06-644-1713
上新電機㈱せっつとんだ店
569 高槻市大畑町24-10／0726-93-7521
上新電機㈱いばらぎ店
567 茨木市双葉町2-18／0726-32-8741
上新電機㈱たかつき店
569 高槻市北園町18-5／0726-85-1991
上新電機㈱ねやがわ店
572 寝屋川市緑町4-20／0720-34-1166
上新電機㈱堺東店
590 堺市北瓦町1-26／0722-22-0950
㈱ライフストア泉南店
590-05 泉南市信達牧野207／07248-2-4001
㈱キディランド大阪梅田店
530 大阪市北区芝田1-1-3／06-372-7701
東亜エレシャック㈱
556 大阪市浪速区日本橋5-11-7／06-644-0111
㈱ダイエー吹田店
564 吹田市朝日町2-101／06-319-1147
大阪バイトショップ
556 大阪市浪速区日本橋東4-4-10／06-632-0207
マイコンショップCSK
530 大阪市北区梅田1 1-3(大阪駅前第三ビルB1)／06 345 3351
西武百貨店八尾店マイコンショップ
581 八尾市光町2 158／0729-97-0111
西武百貨店関西高槻店マイコン売場
569 高槻市白梅町4 1／0726-83-0111
●京都府
アドコム電子
617 京都府長岡京市天神1-8-7／075-939-5231
ヒエン堂
600 京都市下京区寺町通綾小路角／075-361-0371
●奈良
上新電機㈱なら店
630 奈良市三条町478-1／0742-24-3831
●和歌山
コバヤカワ電器
640 和歌山市米屋町9番地／0734-31-3388
㈱和研エレックス
640 和歌山市東高松3-98-2／0734-46-2016
●兵庫
㈱ダイエーパレックス
650 神戸市中央区三の宮町1-6-7／078-391-7911
㈱ダイエー塚口店
661 兵庫県尼崎市南塚口町2-1-1-106／06-427-4401
㈱ダイエー姫路店
670 姫路市中二階町35／0792-84-2151
せいでん本店コンピュータスペース
650 神戸市中央区三宮町1-5-8／078-391-8171
せいでん姫路店
670 姫路市光源寺前11番(星電社姫路本店B1)
／0792 88 1717
せいでん明石店
673 明石市大明石町1-7-4／078-917-5555
小泉コンピュータ㈱
651 神戸市灘区新在家北町1-1-19／078-851-2050
上新電機㈱にしのみや店
662 西宮市河原町5-11／0798-71-1171
上新電機㈱いたみ店
664 伊丹市昆陽池1-69／0727-77-5101
せいでん名谷店
654 神戸市須磨区中落合2-2-3 須磨パティオ3号館
／078-791-8171
●滋賀
西武百貨店大津店マイコンショップ
520 大津市におの浜2-3-1／0775-25 -0111
●鳥取――――――――――中国
米子コンピュータ・システム
683 米子市西福原736-2／0859-34-1200
●岡山
第一産業(倉敷)
710 倉敷市笹沖字汐田1209-1／0864-22-2011
第一産業(岡山)
700 岡山市中山下1-8-15／0862-32-6511
第一産業㈱青江店
700 岡山市青江525-8／0862-25-1911
㈱ダイエー岡山店
700 岡山市駅前町1-8-5-701／0862-32-8811
●広島
第一産業(広島本店)
730 広島市中区紙屋町2-1-18／0822-47-5111
第一産業㈱福山店
720 福山市三ノ丸町6-8／0849-23-1566
●山口
エノモト電子
745 徳山市西辻5762／0834-31-1725
●愛媛――――――――――四国
第一産業(松山)
790 松山市宮田町188 1／0899-33-2311
デジック(松山)
790 松山市本町6-3-7(ロータリー本町ビル1F)／0899-24-0914
●徳島
山菱電子販売
770 徳島市助任橋1-22潤水ビル2F／0886-23-7183
●高知
高知マイコンセンター
780 高知市南御座9-6／0888-84-3750
●福岡――――――――――九州
カホパーツセンター(福岡)
810 福岡市中央区天神2-4 -27／092-714-5155
カホパーツセンター(久留米)
830 久留米市天神町2-44／0942-35-8478
カホパーツセンター(大牟田)
836 大牟田市栄町2 16／0944-52-4367
カホパーツセンター(飯塚)
820 飯塚市吉原町10-7／09482-5-2468
カホパーツセンター(小倉)
802 北九州市小倉北区京町3-6-22／093-551-3688
福岡バイトショップ
812 福岡市博多区博多駅前2-129(扇寿ビル)
／092-713-1298－9
●長崎
㈱テクノソフト
857 佐世保市松浦町2-8田中ビル4F／0956-23-5223
カホパーツセンター(長崎)
850 長崎市油屋町2／0958-21-1079
●熊本
㈱ベーシック・システム
861-41 熊本市御幸笛田町226-11／0963-78-4927
●宮崎
㈱IBC
880-21 宮崎市宮脇町89-4／0985-27-4326
●鹿児島
㈱中村
892 鹿児島市新家敷町16-54／0992-26-9667

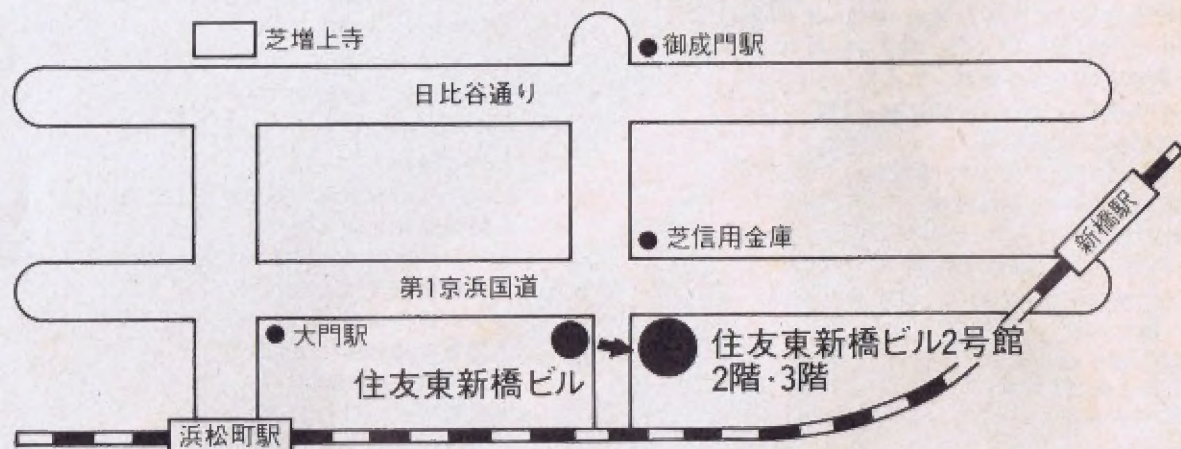
VIC!MESSAGE

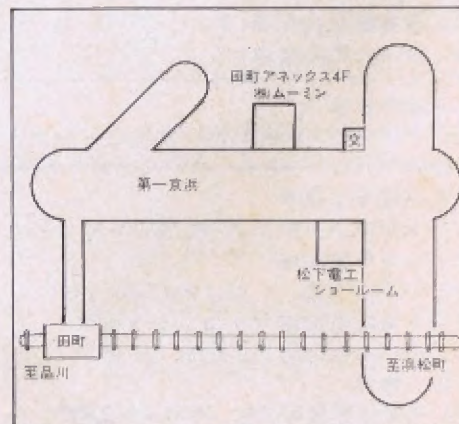
世界へ飛躍するためのコモドール。戦力拡充のため本社を移転！

新所在地
〒105 東京都港区東新橋2-12-7
住友東新橋ビル２号館2,3階
☎03-433-6111（大代表）

ますます意気さかん。期待してね、コモドール。

パーソナルコンピュータのメーカーとして、着実な歩みをつづけるコモドール・ジャパン。その急成長ぶりは業界でも広く知られるところです。年ごとに拡大する業容をみても、それは明らかなもの。
従って、本社事務所が手狭になるのも当然の理で、８月１日をもって新所在地にて業務を開始することになりました。
装いも新たに、エィッとばかりに気合いも入ったコモドール・ジャパンの、一層の飛躍を期待して、いいとも！

"COMMODORE TOWN"
HIGASHI SHINBASHI

芝増上寺
御成門駅
日比谷通り
芝信用金庫
新橋駅
第1京浜国道
大門駅
住友東新橋ビル
住友東新橋ビル2号館
2階・3階
浜松町駅

ヨロシクね。みんなの友だち「ムーミン」。

コモドール・コンピュータの国内販売店、といえば㈱ムーミン。
「カワイイ名前ね」「ボクらの仲間って雰囲気だね」なんて、ただ今、あちこちで親しまれています。
これからも、皆さまのより身近かなところで活躍していきたいと思っています。
コンピュータのことで知りたいこと、聞きたいこと、いろいろあったら連絡ください、こちらまで。
☎03-455-6161 ムーミン・ホットライン

株式会社ムーミン 〒108 東京都港区芝4-7-7 田町アネックス
Tel☎03-455-6161（代表）

BASIC
is Best Communication.

まずBASICを制覇(マスター)する。パソコンとの対話はそこから始まります。

COMMODORE 64

BASICとは、**B**eginner's **A**ll-purpose **S**ymbolic **I**nstruction **C**odeのこと。つまり初心者向けの多目的な命令語を言います。BASICを使いこなせたら、パソコンはグーンと身近かになる。コモドール 64を使いこなせたら、コンピュータをより鮮明に理解できる。コモドール 64は、BASICマスターのマニュアルマシン。持てる限りのノウハウを完全に凝縮したコモドールの自信作です。だから、まずBASICから。だから、コモドール64で。

近未来の夢は、そこから実現します。──さらに ●複雑なテクニックぬきで超迫力グラフィックがプロデュースできるスプライト機能(VICⅡチップ)●最少のプログラムでオリジナルサウンドをらくらく創造。驚異のシンセサイザ機能(SIDチップ)。2大特長はパソコンを知りつくした人にとっても超魅力。

COMPUTER OF THE YEAR
EXPRESS
記念キャンペーン実施中。

'82の最高名誉賞"Computer of the Year"を受賞したコモドール。信頼のハイクオリティ&リーズナブルプライスを全世界に実証する出来事でした。そしてもうひとつ、日本総発売元がmuminに変わりました。これを記念し感謝の意をこめて、ただ今キャンペーンを実施中。コモドール64お買上げの方に①パソコンのA to Zとプログラム作成が簡明に説かれたBASICテキストおよび②テキストに即したデモソフト(サウンド・カラー・グラフィックス・スプライト機能を含む)や、理解しやすい対話式クイズソフト、ドリルソフトなど全17のプログラムを収めたソフトカセット2本、さらに③フローチャート作成用のテンプレートをセットして、無料プレゼント！期間は10月15日まで。好評を博し続行中。お早めにどうぞ。

※いま、英国で10万本以上のベストセラー

●記念キャンペーン実施店は……

九十九電機㈱　上新電機㈱J&P　マイコンショップCSK　パスカル　FBシステム　真光無線㈱　関東電子　㈱船橋そごう3F　コムスポット　共立　カトー無線　せいでん　第一産業㈱　カホパーツセンター　ヒエン堂　コバヤカワ電器　トヨムラ　㈱シンプル

●カタログご希望の方は60円切手同封のうえ、㈱ムーミンまでお申込みください。

commodore COMPUTER

日本総発売元　株式会社ムーミン

〒108 東京都港区芝4-7-7 田町アネックス　TEL (03)455-6161(代)

大阪支社　〒542 大阪市南区長堀橋筋1-45-1 日生長堀橋ビル　TEL (06)251-4001

僕らは、身構えすぎていた。

コストダウンも進歩です。
ここまで来た生産技術。
ニュープライスに挑戦。

主な仕様●使用言語：BASIC, 機械語●CPU：MPS6502A●ROM：20Kバイト標準実装（最大32Kバイトまで拡張可）●RAM：5Kバイト標準実装（最大32Kバイトまで拡張可）●表示構成：ノーマルモード――横22字×縦23行506文字。ハイレゾリューション・グラフィック・モード――176×160ドット●カラー：キャラクター8色、ボーダー8色、スクリーン16色●サウンド：3サウンド+1ノイズ・ジェネレーター内蔵●インターフェイス：カセット・インターフェイス／ビデオ・インターフェイス／ユーザーポート／シリアルポート／コントロールポート／メモリー・エクスパンション・バス内蔵●使用可能ディスプレイ：家庭用TV（RFモジュレーターは本体付属）、専用カラー・モニター（コンポジット入力）●使用電源：AC100V50／60Hz　**定価 49,800円**

commodore japan limited

製造元 **コモドール・ジャパン株式会社** ●東京都港区東新橋2-12-7住友東新橋ビル2号館東京本社〒105☎03(433)6111(大代表)

総発売元 **株式会社ムーミン** ●東京都港区芝4-7-7 田町アネックス　東京本社〒108☎03(455)6161(代表)●大阪市南区長堀橋筋1-45-1日生長堀橋ビル大阪支店〒542☎06(251)4001(代表)

VIC! Vol.11 昭和58年10月25日発行（郵便振替口座番号 東京8-42681番 名称コモドールジャパン） 定価500円